LLNL-TR-721937

LAWRENCE LIVERMORE NATIONAL LABORATORY

Increasing Mission Impact Through Exploratory Target Shots

D. Mathisen

February 7, 2017

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

LLNL-TR-721937

Lawrence Livermore National Laboratory

Increasing Mission Impact Through Exploratory Target Shots

Unclassified

System Engineering Paper

January 31, 2016

Author: David G. Mathisen

Disclaimer

Table of Contents

Section 1

Executive Summary

1.1 Background

1.1.1 What is NIF?

NIF is the world's largest, most energetic laser. It provides the experimental physics community the ability to perform a wide variety of high energy, high density physics experiments. NIF is roughly the size of a football stadium and, in fact, one could place three football fields on its roof. It contains 192 laser beams, each with a spatial aperture of $40cm^2$ with a total capability of delivering 1.8MJ of energy to its target chamber center. NIF's core missions include stockpile stewardship, exploring clean fusion energy, and discovery science.

1.1.2 The existing ecosystem

Given NIF's size and complexity it is not surprising that it is supported by a mature and well defined set of subsystems and procedures. It is important that the solution leverage our current tool sets and provide a workflow and output in keeping with that which is already well trained and understood by a distributed community. NIF is a system of systems. Understanding the roles of those systems, their active stakeholders, and reasonable boundaries for change make those systems themselves active stakeholders to be considered.

At present a standard NIF Experiment defines a single target and its backup. All participating beams and diagnostics are assumed related to the goals of the experiment. NIF's default shot execution behavior is to prepare for and then fire at a single target in the course of a Shot Lifecycle. Each Shot Lifecycle may take from four to eight hours to execute, a significant portion of which is related to aligning and preparing bundles to fire.

1.1.3 Understanding the customers' needs

Providing the ability to conduct a greater number of target shots allows us to explore a greater range of physics parameters. Doing so even more efficiently further improves upon that potential. By meeting the core mission statement, "**The NIF experimental physics user community needs to more efficiently perform series of exploratory target experiments**," we would allow a NIF User to quickly explore a range of parameters in order to better refine the parameters of their full NIF Shots, resulting in greater mission impact.

1.1.4 Refining customers' needs through a prototype

As a means of better understanding the potentials of this project and our customers' needs, a prototype referred to as "Gatling 1.0" was developed. By combining changes to the Integrated Computer Control Systems' (ICCS) shot automation behavior and development of detailed NIF procedures, a process emerged which allowed for executing four target shots in the course of a single Shot Lifecycle.

A standard Shot Lifecycle for NIF includes one or more Rod Shots in which we verify the energy provided by our pre-amplifiers. After each Rod Shot we would return to a "Ready" state at which time we could select to run another Rod Shot or select to fire a System Shot. A System Shot involves all participating beams of the laser using both the pre-amplifiers and main amplifiers. Having fired all beams to the target Shot Execution would automatically safe systems, collect data and end the Shot Lifecycle.

The Gatling 1.0 prototype allows us, using similar targets, to take four target shots during a single Shot Lifecycle. This is done by preparing the entire laser composed of 24 bundles of eight beams each and then selecting specific bundles to be fired to the installed target. Rather than end the shot, it then returns to a "Ready" state and allows operations to change targets and fire another set of bundles without repeating the costly laser setup activities.

Delivery of a working prototype required that we consider the existing architecture and limit changes in a way that allowed for a rapid delivery with minimal development impact. The reference architecture, to be discussed in greater detail later, has three high level conceptual components; ***Shot Planner***, ***Shot Execution***, and ***Data Visualization***.

The prototype minimized changes in the Shot Planner area which were limited to administrative management through spreadsheets. Most software changes for the prototype were made in Shot Execution which was modified to allow firing of selected bundles during a standard system shot, a return to "Ready" state following a system shot, and increment the Shot ID to allow the NIF User to recognize their data from each of the multiple system shots. This relied heavily on detailed manual procedures and operator expertise to allow for and describe swapping of targets and selecting of bundles. Data Visualization changes were relatively limited and included recognizing the newly indexed System Shot IDs.

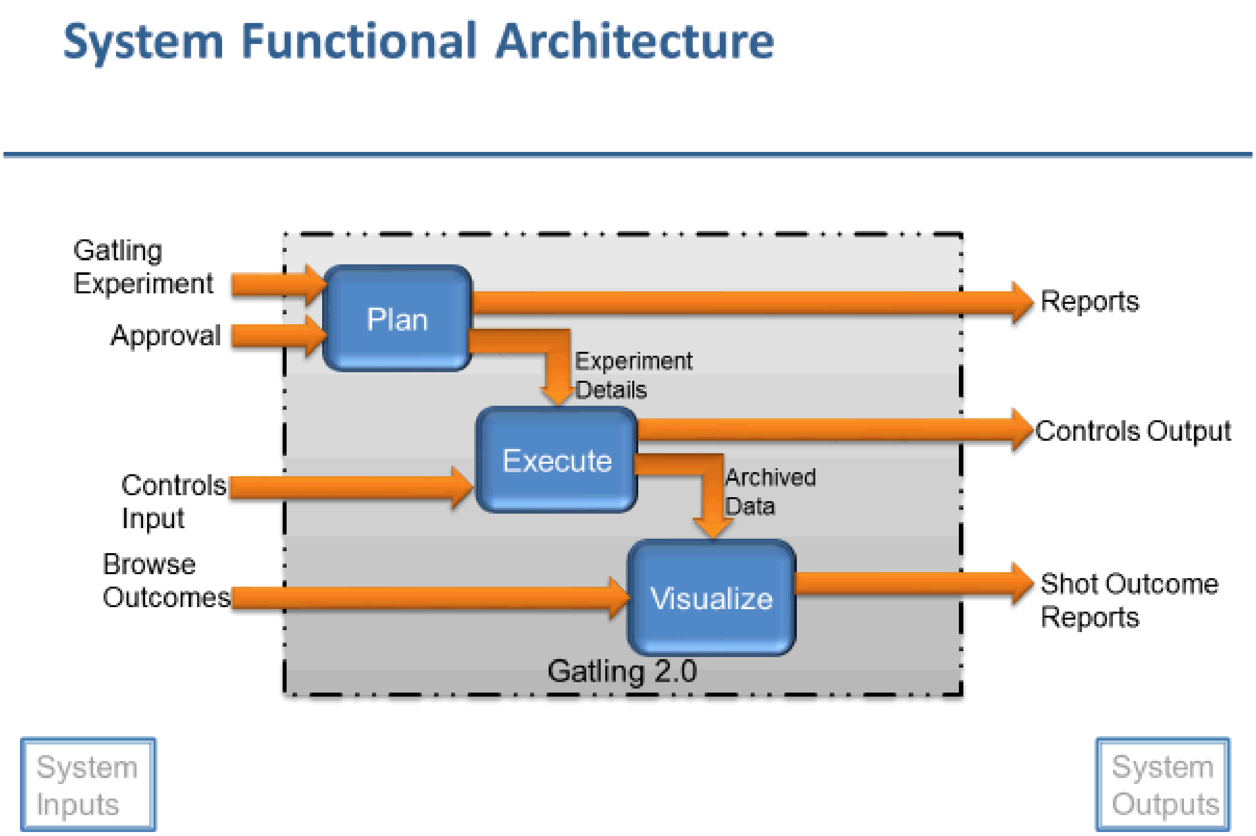
1.1.5 Scope of document

Through the prototype's use, the concept's potential was proven. Use of the prototype also provided insight to the prototypical shortcomings. **This document describes the efforts related to addressing the needs of all stakeholders and developing the "Gatling 1.0" prototype capability into a mature product referred to as "Gatling 2.0."**

1.2 Overview of solution

While our primary stakeholder is the NIF User seeking the capability, achieving it requires the input from a range of other stakeholders: from *shot planner* to *shot execution* and post shot *data visualization*.

It also requires an understanding of current procedures and operational norms which relate to the roles and interplay of this range of stakeholders. It is important that the defined need be met wherever possible by leveraging existing capabilities of both the toolsets involved and skillsets of those using the toolsets. To this end, both active users and internal system owners have been made partners in analyzing the integrated approach and its impacts.



While the functional architecture appears unchanged at the top level as compared with the "Gatling 1.0" prototype, the modifications required to achieve the mission statement "**The NIF experimental physics user community needs to more efficiently perform series of exploratory target experiments,"** are distributed within these sub functions.

Plan has been updated to provide simple bulk input to users without impacting traditional users and provide validation and ordering of component experiments. This is done by adding the "Experiment Editor" which allows a NIF User to submit a set of place holders to "Shot Planning" and assures their relation to one another and indication that they are part of a Gatling Sequence. The NIF Shot Scheduler (person) and Expert Group reviewers may now review upcoming shots with an understanding that those marked Gatling which are related, and consider that in his/her scheduling. After these experiments are further defined the "Gatling Shot Composer" will validate, aggregate, and order them into a "parent" experiment which may be loaded into ICCS Shot Control for execution.

Execute has been updated to leverage ordering information to automate firing and archive as fired. This is done by executing a Parent Experiment which allows for preparation of the entire laser and, in parallel, aligning the first of up to four targets. A rod shot on all bundles is conducted as normal, after which each system shot will fire only those bundles associated with its target as defined in that target's component experiment. Association of targets, bundles and diagnostics is a natural outcome of the system shot. Automation aids the NIF Operator in automatically dropping fired bundles and "re-aggregating" the parent experiment with the target and diagnostic information from the next component experiment.

Visualize has been updated to browse the Gatling sequence and view any Gatling fired shot as if fired standalone. This is achieved using a combination of the archived data, machine history, and Gatling Shot Sequence information to allow either browsing from the parent experiment or viewing of a single component (including its preparation and rod shots, as if it had run standalone).

Systems requiring new interfaces and functionality are included in the detail of this report.

This modification to NIF is driven solely by mission impact. By providing the NIF User community an opportunity to more readily execute exploratory target shot series, the breadth of exploration is increased, allowing us to more quickly identify those parameters which hold the most promise.

Section 2

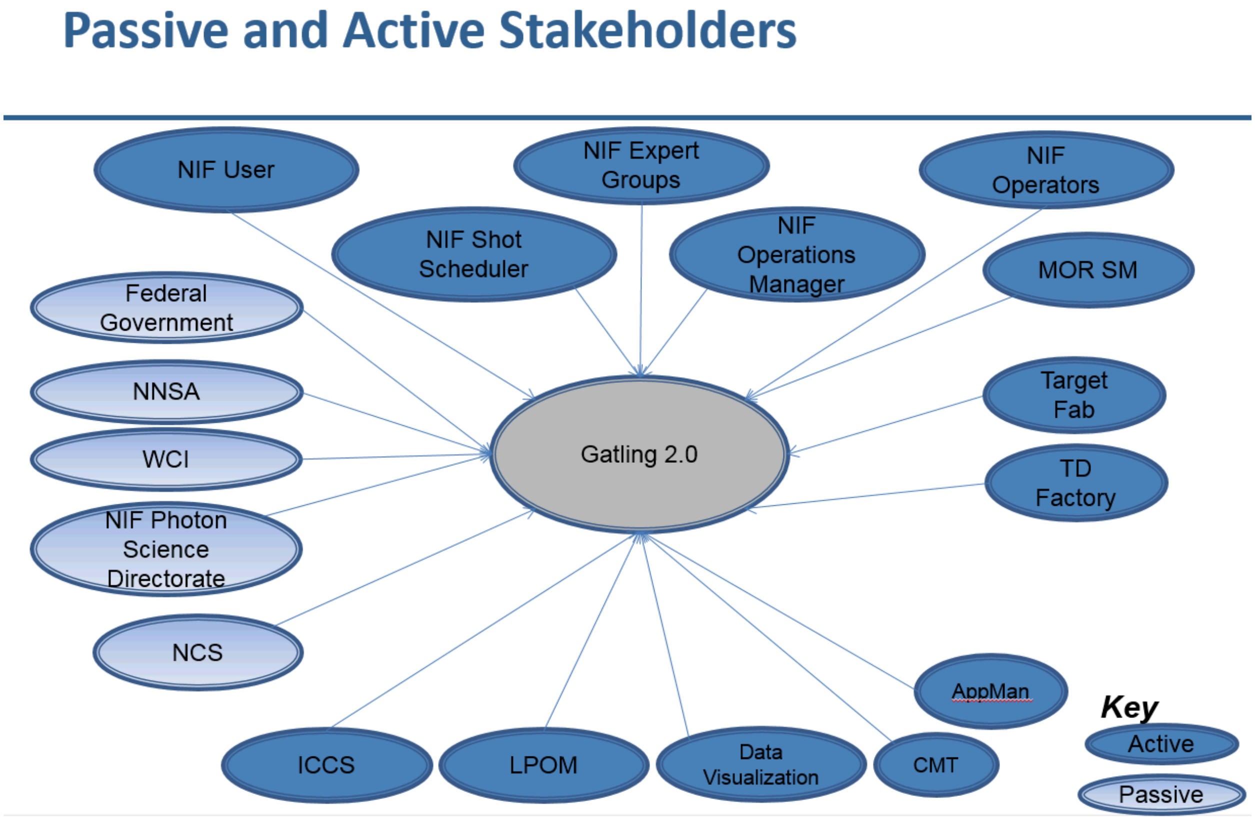
Mission Description

2.1 Mission Statement

The NIF experimental physics user community needs to efficiently perform series of exploratory target experiments.

2.2 Stakeholders

2.2.1 Stakeholders Overview



2.2.2 Stakeholders: Roles, Viewpoints and Points of Contact

NIF User (active),
  Provide greater experimental capability on NIF
NIF Shot Scheduler (active),
  Understand impacts on shot rates and view potential Gatling candidates
NIF Expert Group Reviewers (active),
  Streamline review and approval
NIF Operations Manager (NOM) (active),
  Experiment approval, minimize facility complexity
NIF Operators (active),
  Shot Director, reduce complexity and maximize reliability
Master Oscillator Room System Manager (MOR SM) (active),
  No changes to existing Fiber Delay Backlighter (FDBL) Reports and tool sets.
Target Diagsnostics (TD) Factory (active),
  Be aware of work load / pace
Target Fabrication (active),
  Be aware of work load / pace
ICCS: Shot Control (active)
  Receive clear, concise, testable requirements
NIF Laser Performance Operations / Modeling (active),
  Provide ability to update target mass information
Data Visualization (active),
  Minimize impacts on Data Visualization developers and users
Campaign Management Tool (CMT) (active)
  Minimize product impact, clear, concise, testable requirements
Approval Manager (AppMan) (active)
  Minimize product impact, clear concise testable requirements
NIF Control Systems (NCS) / Maintenance (passive),
  Reduce subsystem impacts, execute with maintenance in mind, produce clear, concise, testable requirements
NIF & Photon Science Directorate (passive),
  Maximize Mission Impact
WCI (passive),
  Maximize Mission Impact
NNSA (passive),
  Maximize Mission Impact
Federal Government (passive),
  Maximize Mission Impact

2.3 Capabilities and Characteristics

NIF User:

Needs access to more exploratory target shots
Needs a system which provides full system shot energies
Needs a system which provides familiar shot results (characteristic)
Needs a system which leverages existing shot defining workflows (characteristic)
Needs a system which generates target physics data on every shot
Needs system to leverage existing CMT work flows (characteristic)
Needs 'As fired' archived data to reflect selected bundles to avoid causing analysis to fail to process (manual intervention required)
Needs a clear review of "Component Experiment" results as if executed standalone (characteristic)

NIF Shot Scheduler:

Needs to see Gatling experiments from the start to support planning

NIF Expert Group Reviewers:

Need efficient approval methods
Need a system which does not deviate from current approval disciplines
Need to aggregate and/or distribute approvals related to component experiments.\
Need to manage and distribute parent to component and component to parent approval as to improve and clarify Gatling Shot approvals

NOM:

Needs to optimize use of NIF resources
Needs to work with existing staffing levels (characteristic)

MOR SM:

Needs existing toolsets such as FDBL reports and procedures to work seamlessly

TD Factory:

Needs awareness of workload impacts

Target Factory:

Needs awareness of workload impacts

NIF Operators:

Need a simplified (automated) shot execution
Need capability to drop experiment in sequence if automated
Need simplified checklists / reduced manual efforts
Need automated selection of bundles, and positioners/diagnostics from shot to shot
Need automated bundle to target validations
Need automated validation of non-participating bundles configuration in Gatling system shot (shutters, Vacuum Isolation Valves (VIV), positioners, Diagnostic Instrument Manipulators (DIM), etc.)
Need automation of target alignment process at Ready State (including multiple Target Positioners (TP) within series)
Need integrated status verifier validation of system configuration for shot
Need existing toolsets such as FDBL reports and procedures to work seamlessly

NIF Laser Performance Operations / Modeling:

System needs to support retrieval of Target Mass from shot archive

Other Existing internal systems (AppMan, CMT, Data Visualization, ICCS):

Need maintain existing core capabilities (characteristic)
Need to provide capability clearly without distracting or confusing traditional users
Need to fit within schedule and budget constraints (characteristic)

2.4 Key Stakeholder acceptance criteria and sacred expectations.

Provide Higher Shot Rate for exploratory target shots
Provide full system shot energies
Generate Target Physics Data on each shot
Don’t break the existing system
Minimize user impact

Section 3

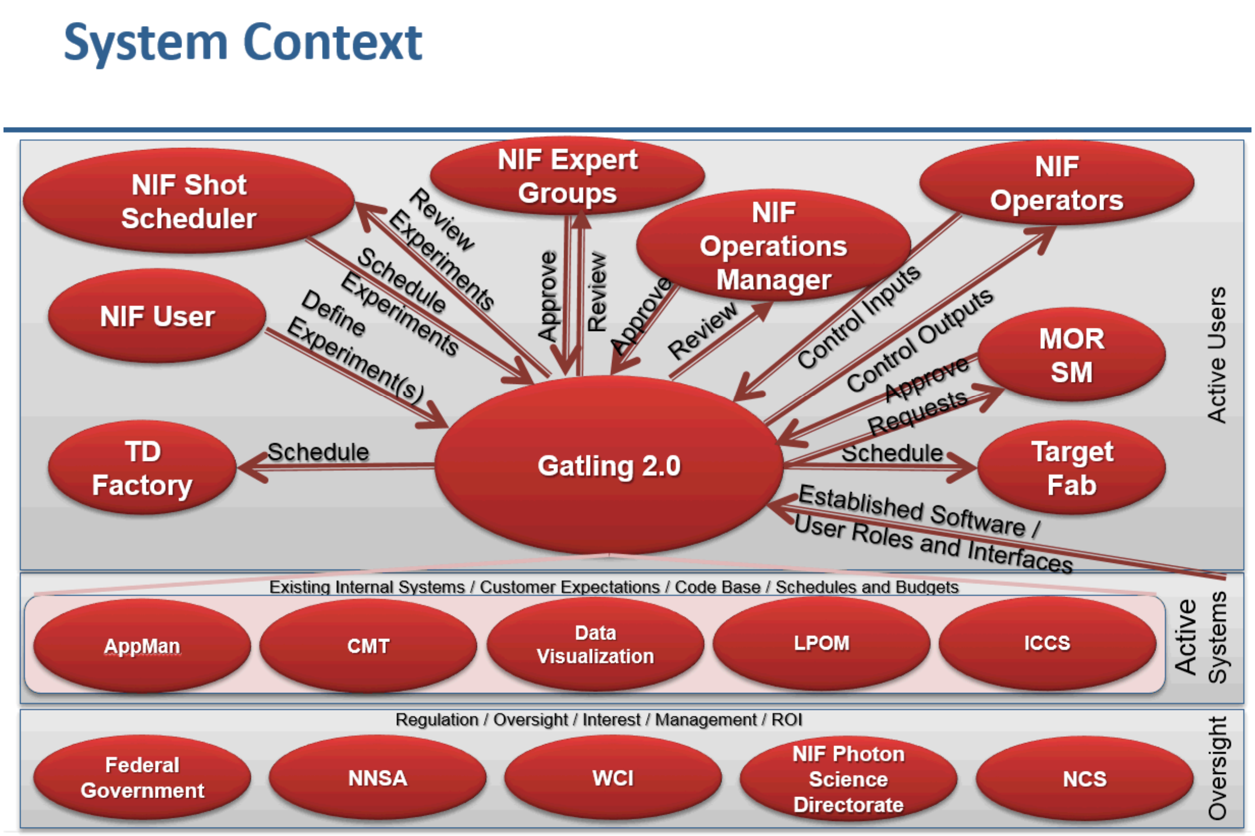
System Definition: Understanding the solution environment

3.1 Context

3.1.1 Context Description

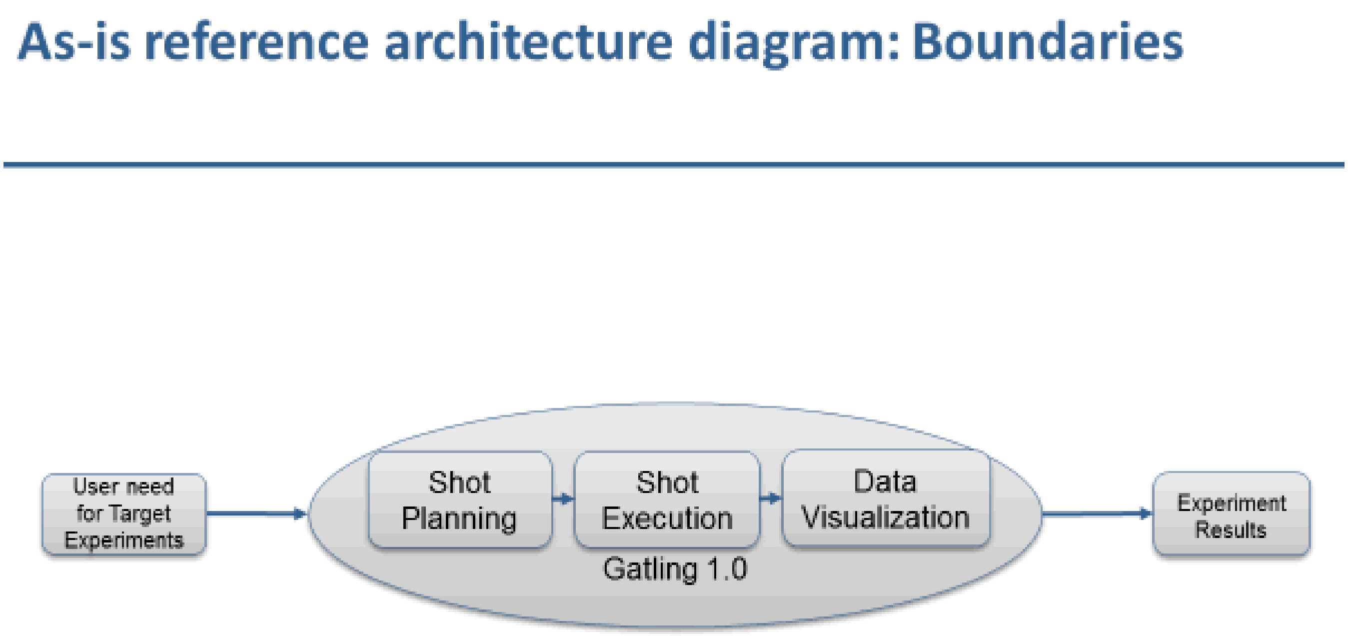
NIF is a system of systems and, as such, carries with it a mature ecosystem. Each of the systems have their own user base and conceptual behaviors. This context diagram shows two forms of active users: those physically interacting with the established system, and "subsystem" owners whose job it is to assure the cohesiveness of their systems, relationships with their customers, and ability to handle code and interface changes required within a shared schedule.

3.1.2 Context Diagram

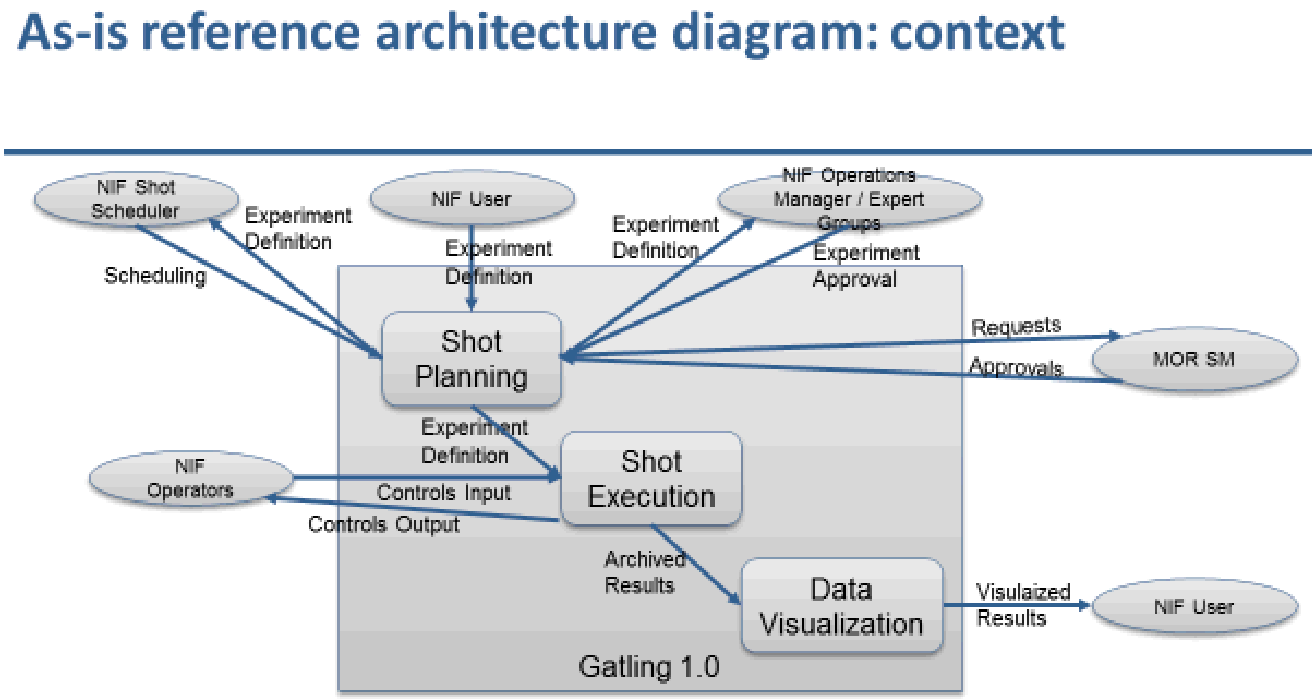


3.2 Reference Architecture

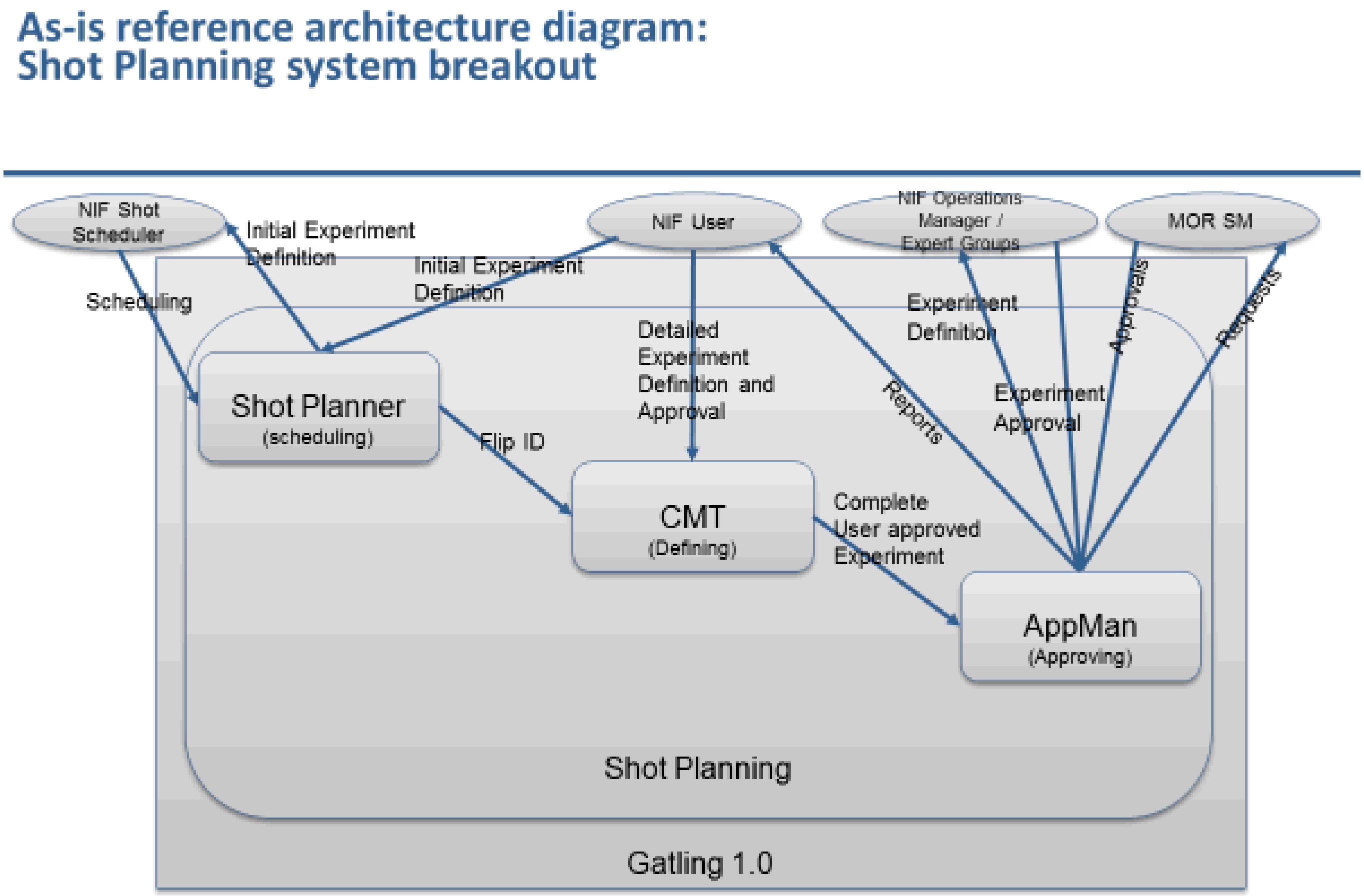
3.2.1 Reference Architecture: Boundaries



3.2.2 Reference Architecture: Context



3.2.2.1 Reference Architecture: Shot Planning System Breakout



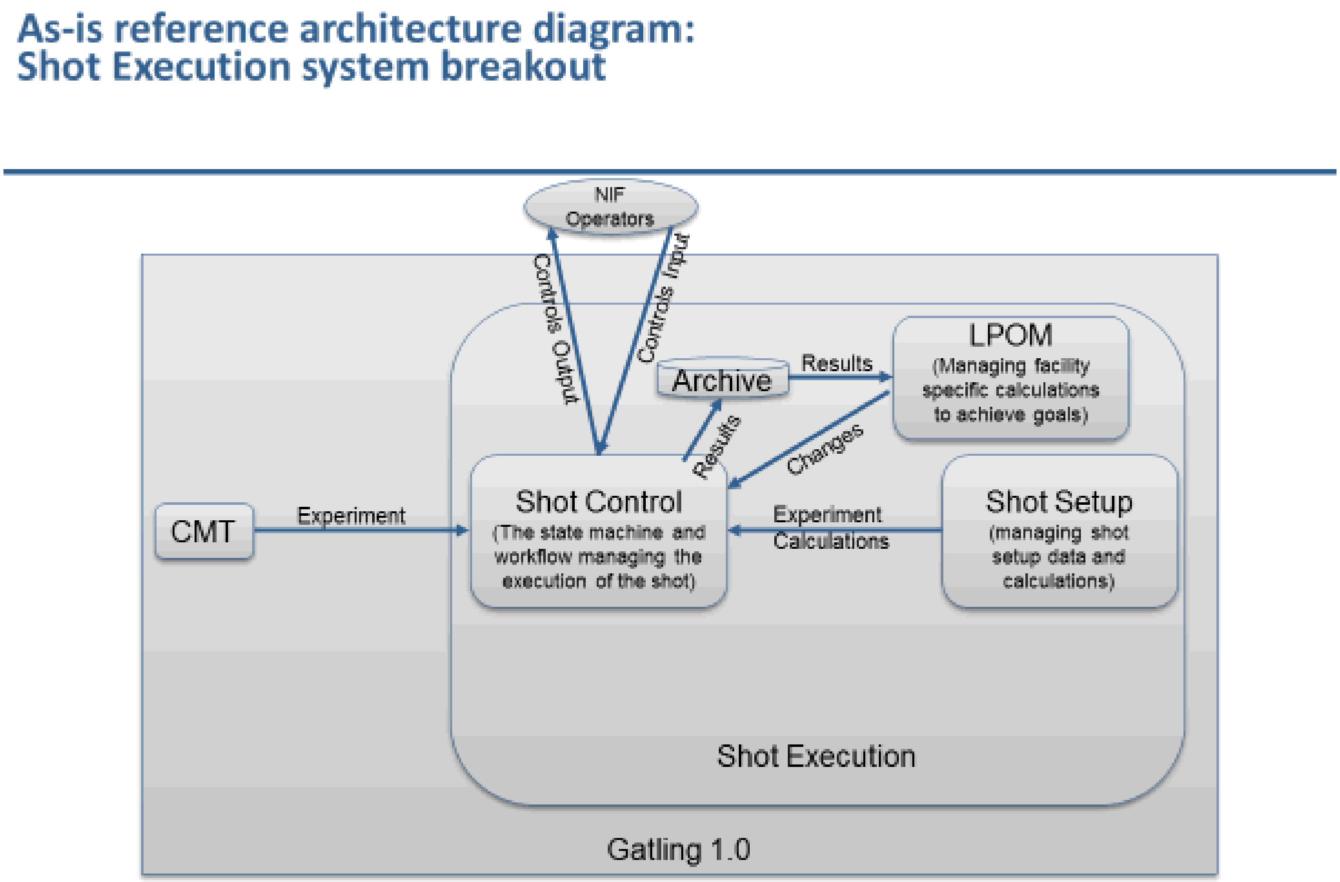
3.2.2.1.1 Reference Architecture: Shot Planning as is

Shot Planning involves developing initial descriptions of upcoming experiments in order to begin allotting schedule time to their execution. As the experiment becomes better defined the initial aliases describing them become bound to fully realized experiment definitions. The NIF Shot Scheduler uses Shot Planner to allot upcoming facility time. Once submitted by the NIF User (or Shot Responsible Individual (RI)) it is then reviewed and approved by various expert groups. With all expert groups having approved, the NOM may then approve the experiment for execution in the NIF Facility.

At this point there is no concept of “Gatling” as a shot type per se—only the expertise of the NIF User / Shot RI to include backup targets to convey target information. Both the scheduler and expert groups must manually be made aware that this experiment represents four target shots. Operations must be fed procedures to align their bundle selections, diagnostics, target installs, and firing order.

The proposed solution would aid in recognizing Gatling target experiments, scheduling them, defining them, and approving them.

3.2.2.2 Reference Architecture: Shot Execution system breakout



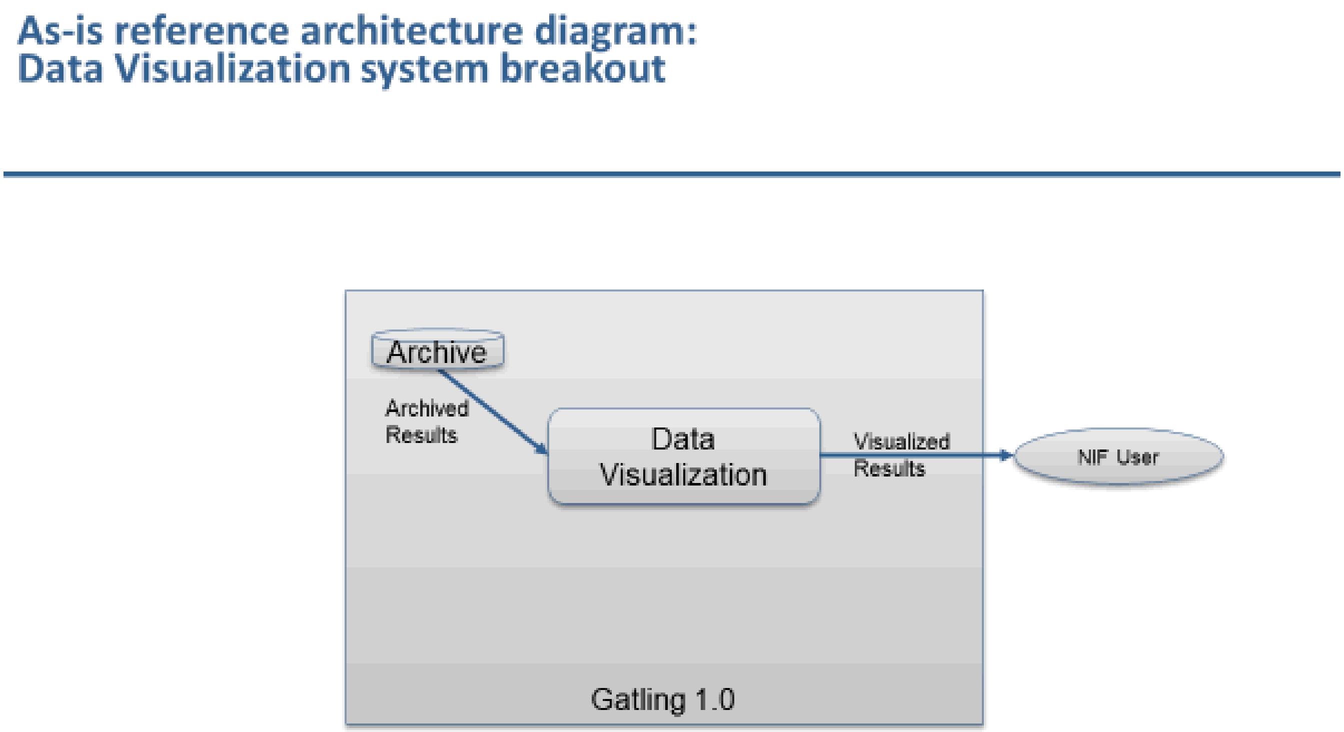
3.2.2.2.1 Reference Architecture: Shot Execution as is

Shot execution involves loading the shot as defined by CMT and executing the shot using ICCS Shot Control, to perform the automation and Shot Setup, to aid in calculating settings and applicabilities based on the experiment goals as requested by CMT. As the shot is executed, information and results are archived. Using this information Laser Performance Operation and Modeling (LPOM) may make suggestions based on initial results for changes to aid in more closely achieving the experiment's goals. Information archived during the shot is also used to update LPOM's view or "Model" of the facility, and inform future behavior.

It is in Shot Execution that the core changes related to the Gatling 1.0 prototype were made. This included modifying the shot Lifecyle to return to the Ready State after having fired a system shot and using a Shot Lifecyle ID of 000 rather than that of the System Shot "999" to allow for multiple system shots to relate to the basis 000 Shot Lifecycle ID. However, swapping of targets and those target details were achieved using "backup target" definitions, detailed procedures and on-hand expertise. In addition, the "as fired" shot results do not reflect the selected bundles, which results in analysis failures.

The proposed solution would aid in automating preparation and execution of the four target experiments in their defined order.

3.2.2.3 Reference Architecture: Data Visualization system breakout



3.2.2.3.1 Reference Architecture: Data Visualization as is

Data Visualization involves presenting the experiment results to the NIF User or Shot RI. This allows a user to lookup by all the results associate to their experiment by either Experiment ID or Shot ID.

Due to the ad hoc method of execution the NIF User must look up his/her results by navigating to the Experiment ID and then reviewing those Rod Shot and System Shot IDs associated with it. Details such as reviewing the target diagnostics fired and bundles used are left as an exercise for the NIF User.

The proposed solution would allow a NIF User to view each Gatling Experiment as if it had been executed as a standalone, including diagnostics and bundles fired.

Section 4

Operational Scenarios

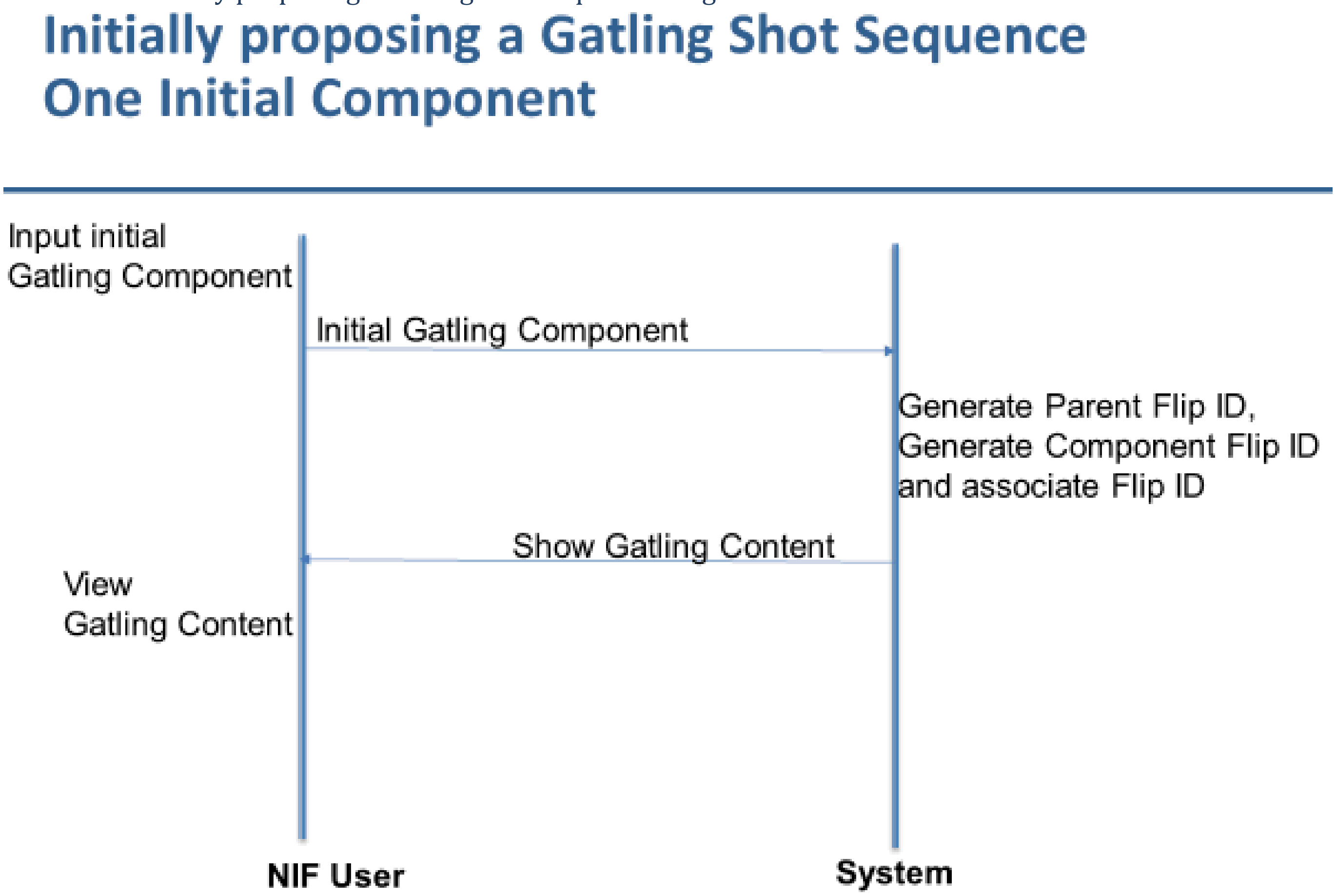
4.1 Primary Operational Scenarios

4.1.1 Shot Planning

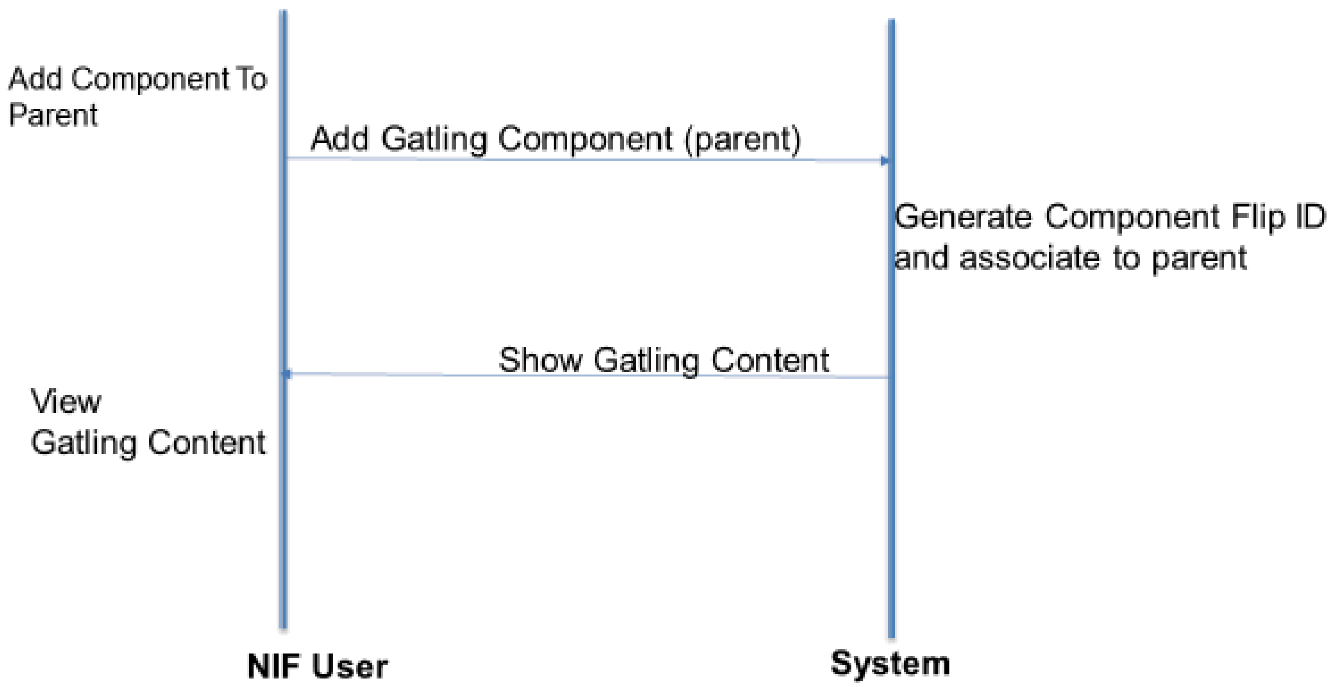
4.1.1.1 Initially proposing a Gatling shot (T –6 months)

When a **NIF User** is interested in scheduling an experiment, they establish a place holder for that experiment in Shot Planning. This provides a rough description of the experiment to be used for preliminary scheduling purposes. The lack of specific experimental detail here is due to the length ahead of time that facility time must be planned. This may be up to six months prior to the shot's execution during which time the NIF User or Shot RI will continue to adjust and improve the parameters of his/her experiment. This should include a preliminary representation of each target shot involved including whether or not it is part of an upcoming "Gatling sequence." This intial representation is represented using what is referred to as a Flip ID.

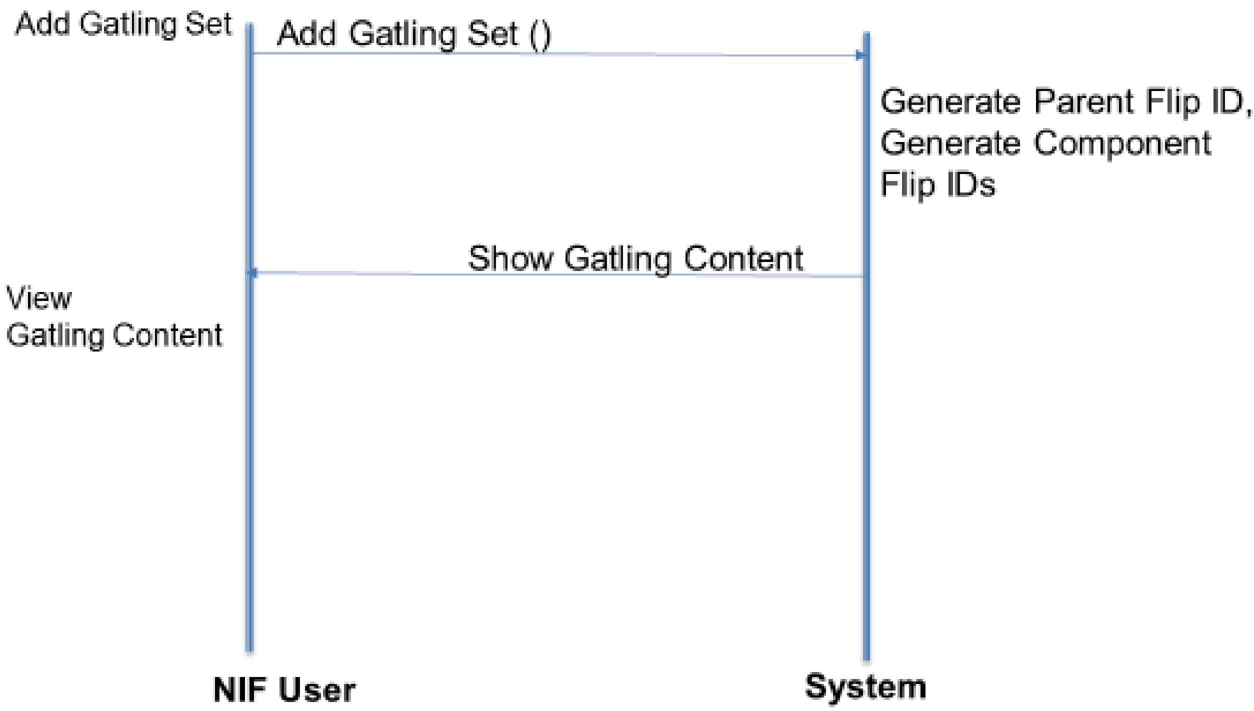
4.1.1.1.1 Initially proposing a Gatling shot Sequence Diagram



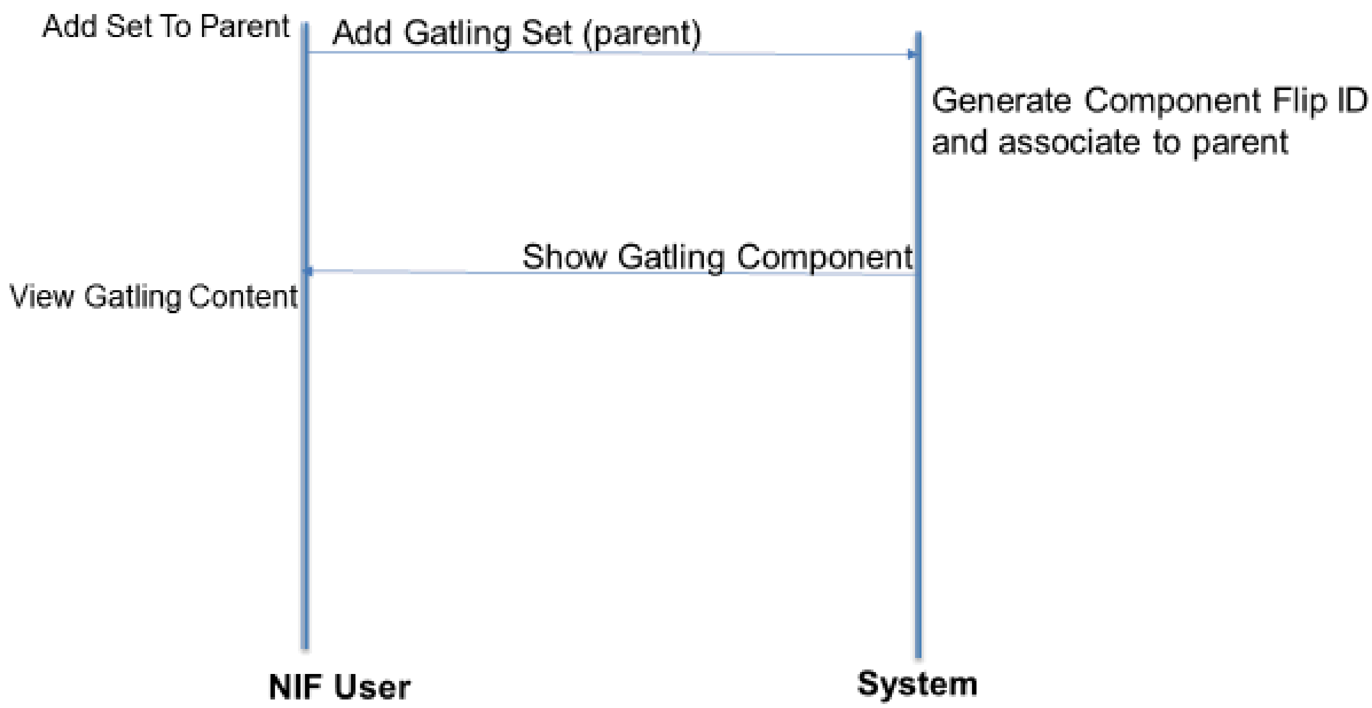
Initially proposing a Gatling Shot Sequence One Additional



Initially proposing a Gatling Shot Sequence Initial Full Set



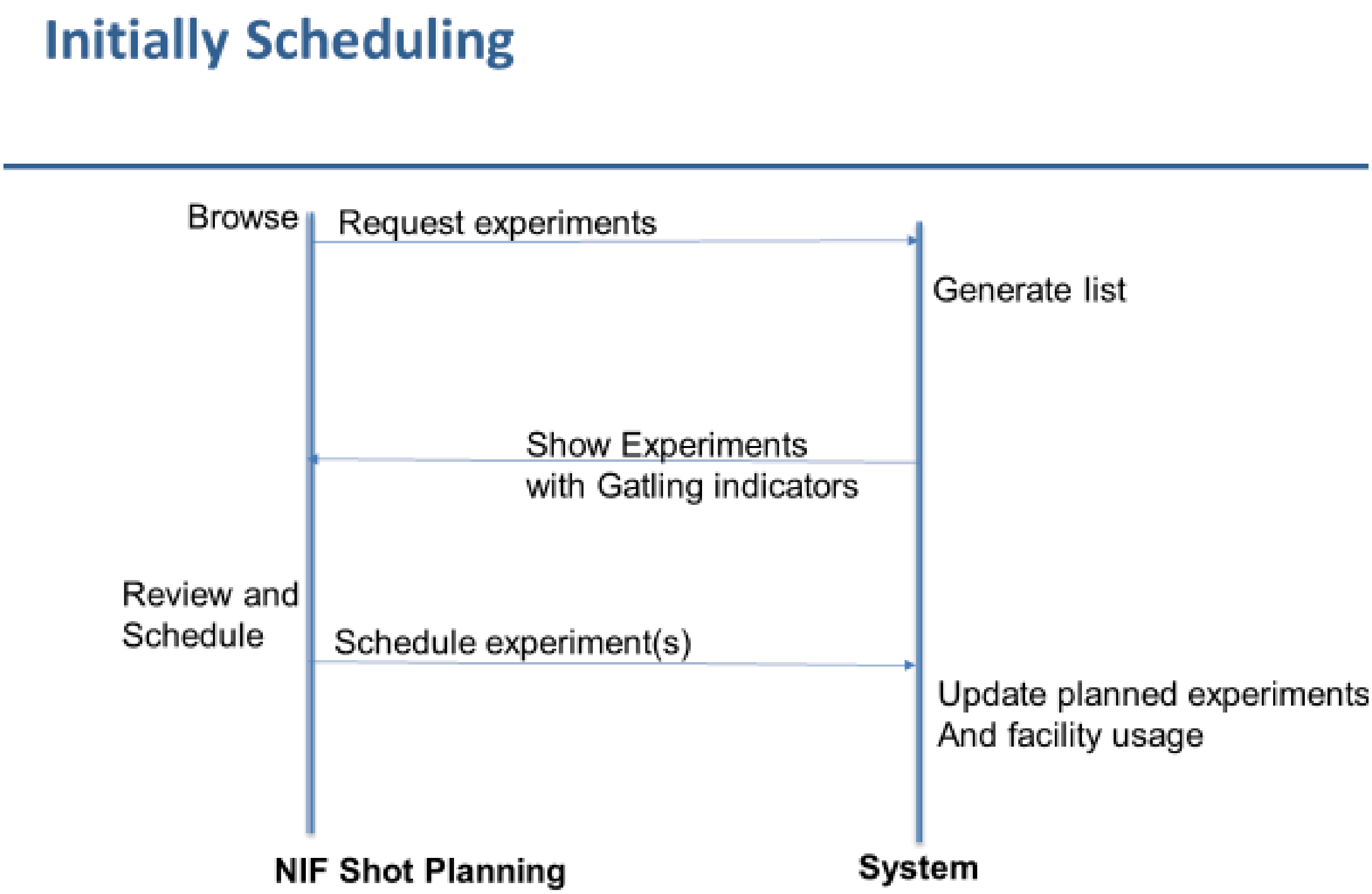
Initially proposing a Gatling Shot Sequence add to Existing Set



4.1.1.2 Initial Scheduling (T –6 Months)

The NIF Shot Scheduler using Shot Planner reviews upcoming shots and considers a broad range of factors in scheduling them. A combination of expertise and facility awareness is used to coordinate shots to meet customer schedules while also smartly considering physical constraints, downtimes, device availability and shot to shot interactions in order to maximize facility productivity. It is important during this phase that the NIF Shot Scheduler have the capability to recognize shots intended to be executed within a Gatling Shot Sequence in order to properly allot time and consider all the resources it will require.

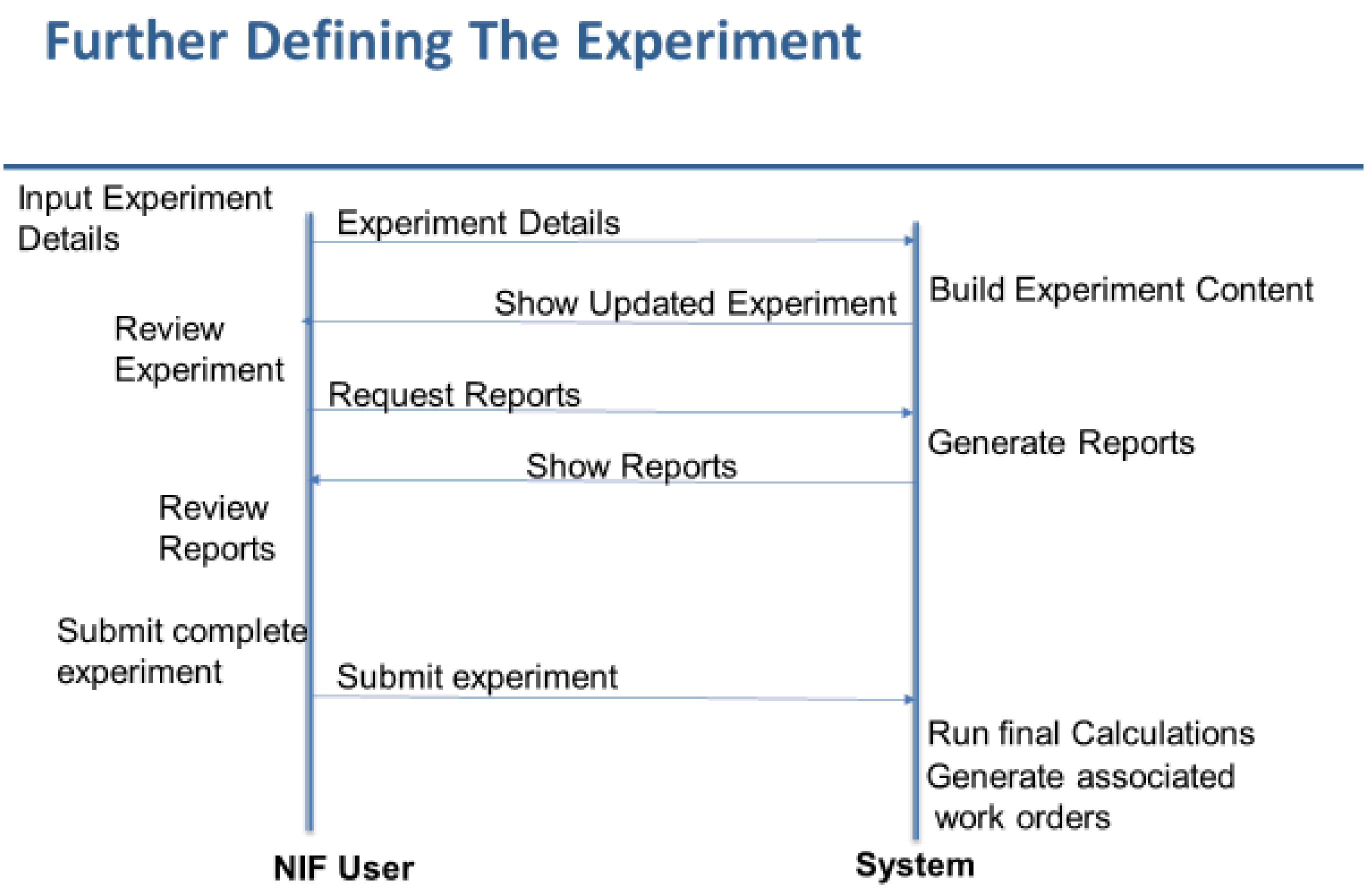
4.1.1.2.1 Initial Scheduling Sequence Diagram



4.1.1.3 Further Defining a Gatling shot (T -6 months – T -1 week)

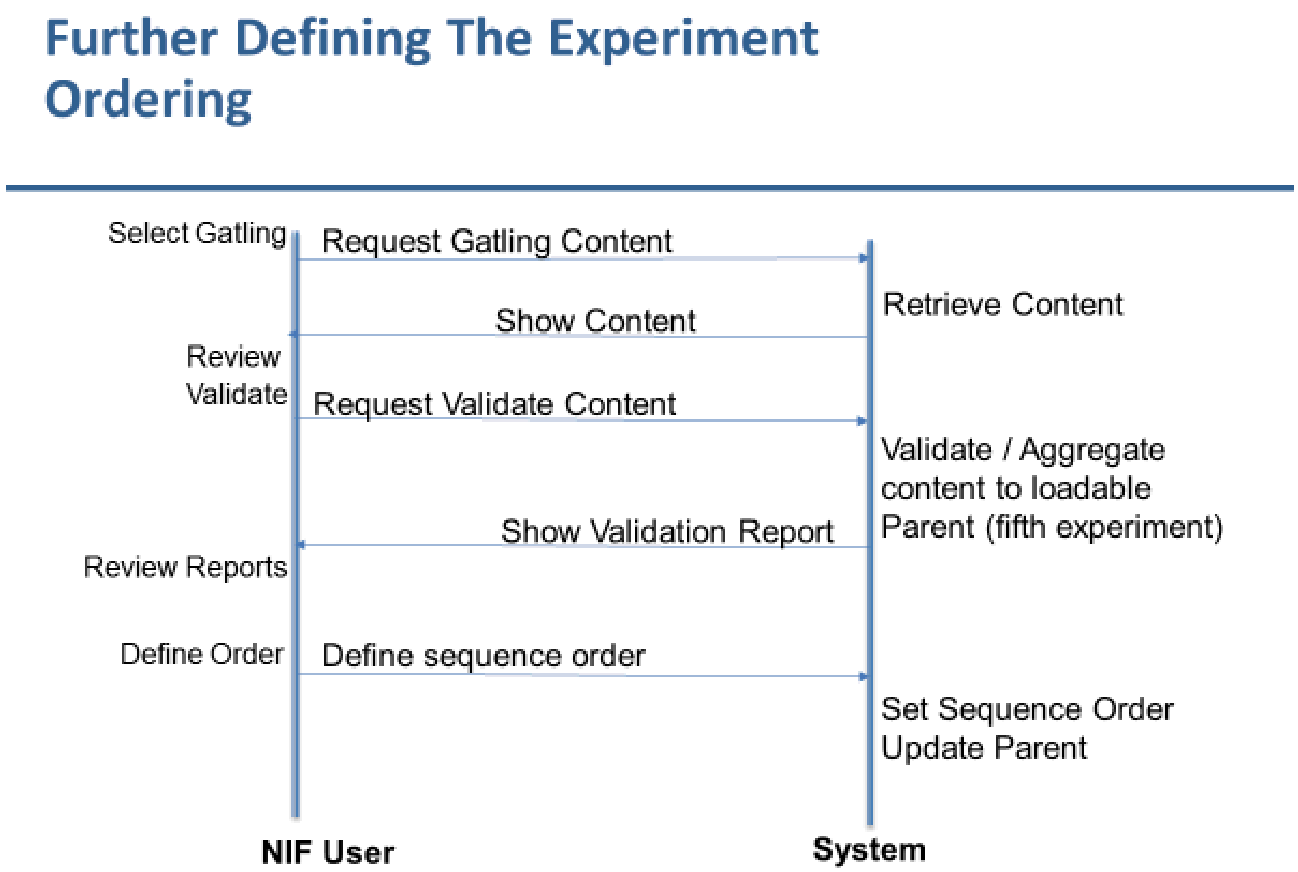
The NIF User interacts with the CMT to define the detailed laser setup and diagnostic usage related to each Shot. During the planning phase each place holder alias was defined and, if intended to be, marked as being part of a Gatling Shot Sequence to allow the NIF Shot Scheduler to allot time appropriately. Here the details regarding bundles, diagnostics and target configurations are defined. CMT allows the user to define every last detail of a shot's setup and execution. This includes but is not limited to reviewing potential work orders which might be generated by requesting this experiment, such as the need for FDBL Fibers. This detail is included as an example of the need to maintain an experiment view such that existing toolsets and procedures remain viable. Once satisfied with both the definition of the experiment, and impact of the work orders it will generate, the NIF User will submit his/her experiment for review and approval.

4.1.1.3.1 Further Defining a Gatling shot Sequence Diagram



4.1.1.4 Further Defining a Gatling shot order (T -6 months – T -1 week)

Here the NIF User must be allowed to define the order of their component experiments. This will of course be used by Shot Automation but is also necessary for Expert Groups to render their approvals.

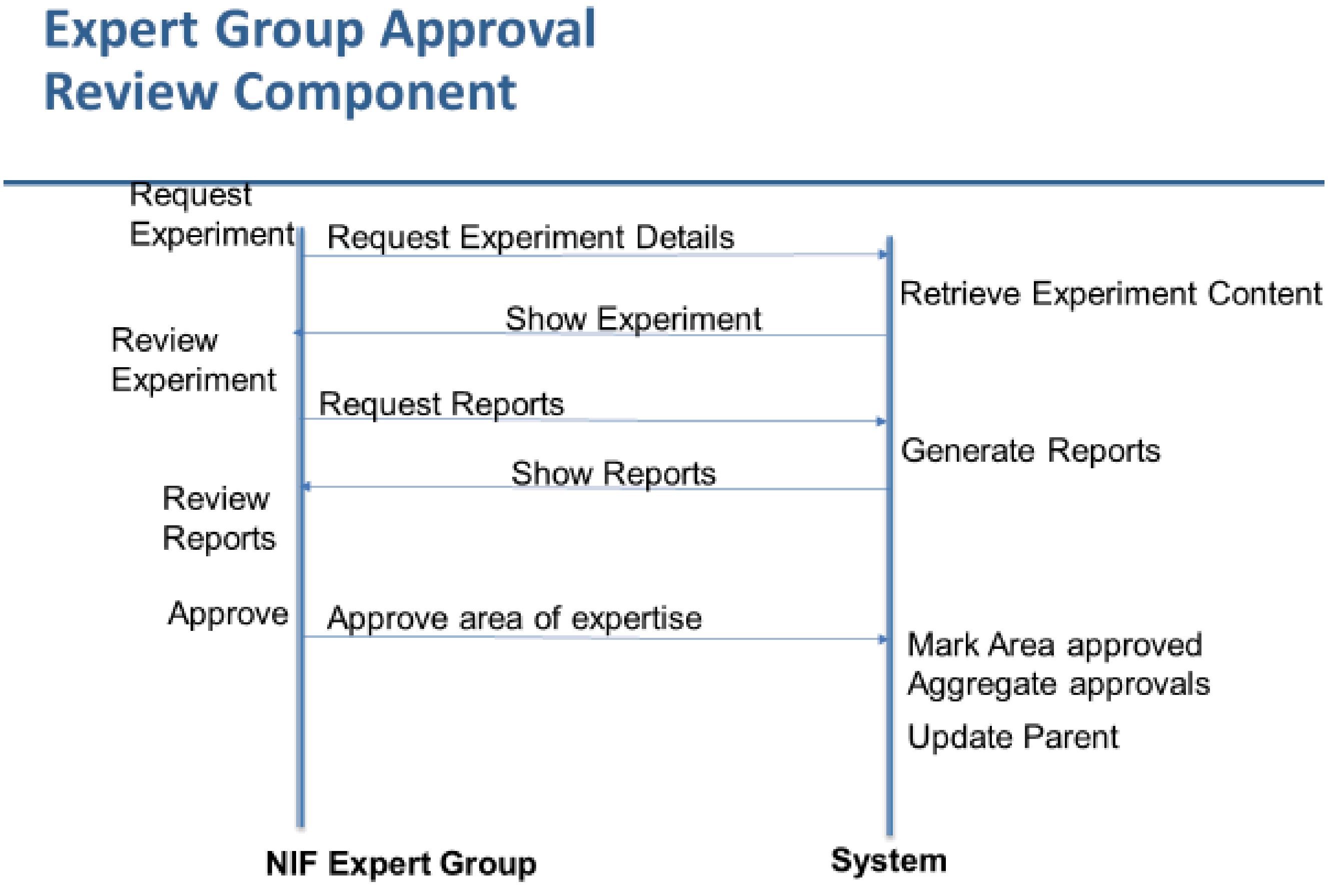


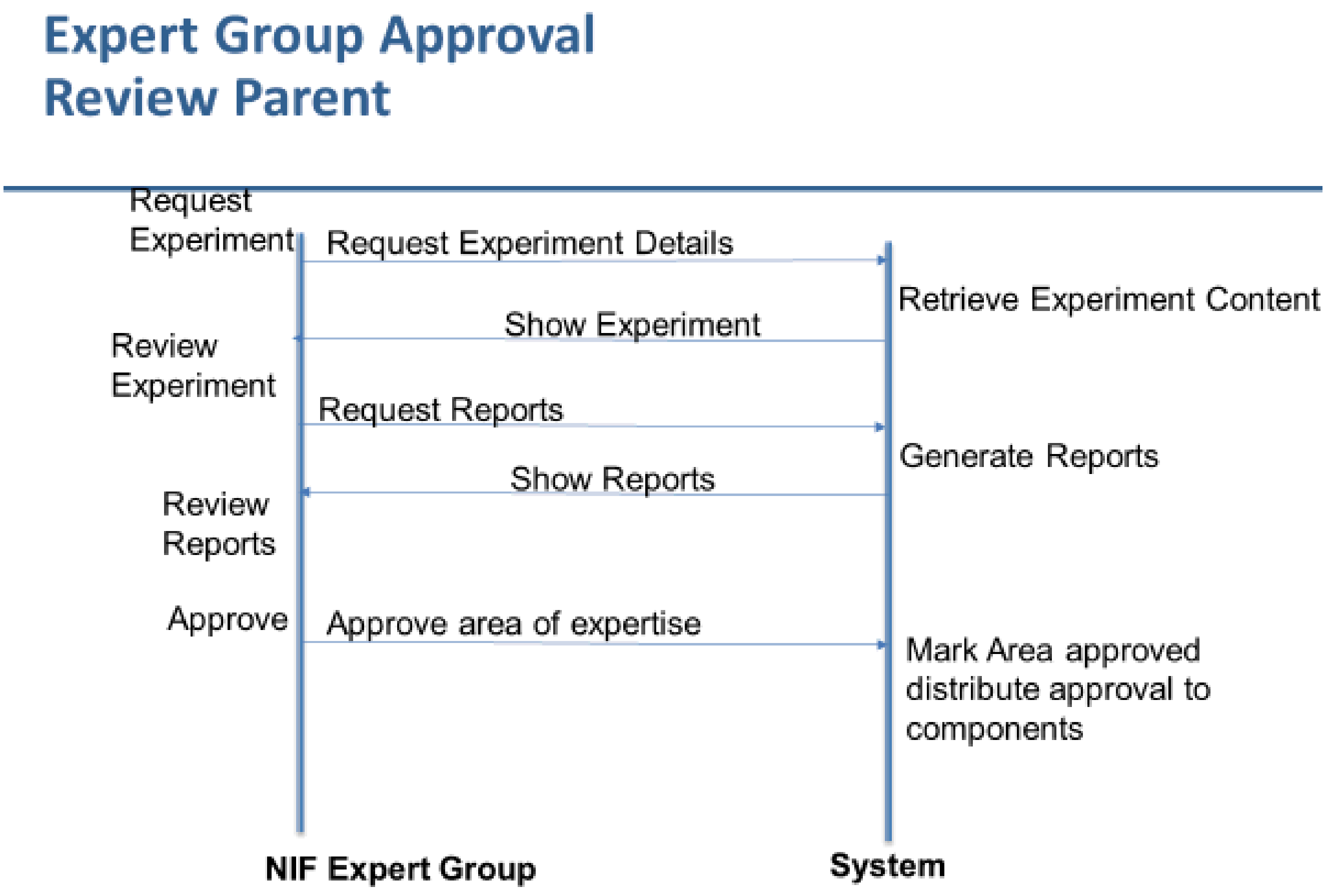
4.1.2 Approving and final scheduling (T -1 month – T -1 week)

4.1.2.1 NIF Expert Group Approvals

The NIF Expert Groups uses various tools including CMT and related LPOM calculations to review the proposed experiment and approves his/her expertise area via Approval Manager (AppMan). For instance, one expert group may be specifically concerned with all things laser setup and therefore reviews to assure that the parameters requested are both effective and safe for the laser itself to execute. It is only when all expert groups have approved all component experiments that the larger Gatling Sequence may then go on to the NOM to grant full approval to be run in the facility. NIF Expert Groups have asked that where possible review and signoff be optimized regarding Gatling Shots. For instance, laser setup may be reviewed in a single experiment aggregated from component experiments and target details may be reviewed and approved in each component experiment. Approval for a Gatling Sequence Experiment is a summation of all of its component experiments.

4.1.2.1.1 NIF Expert Group Approval Sequence Diagram

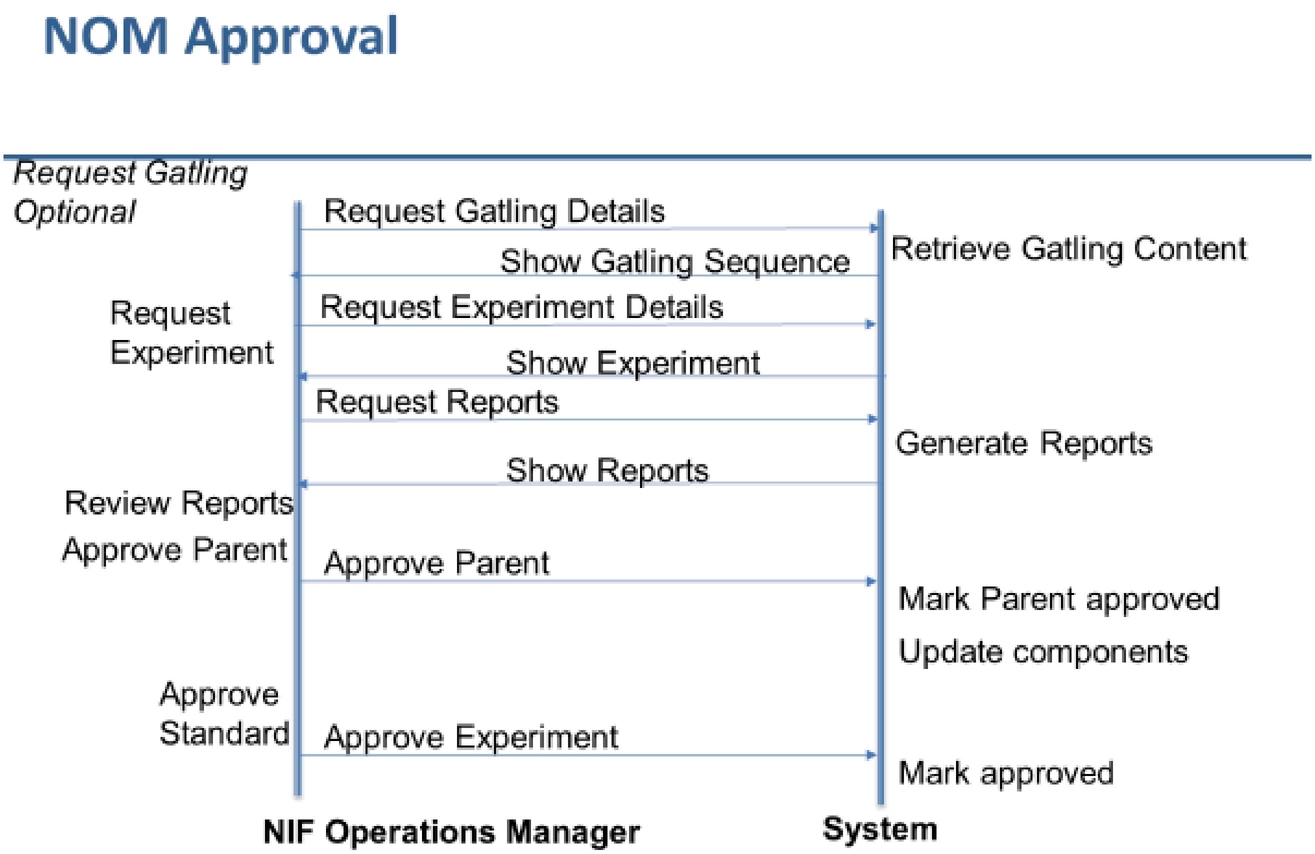




4.1.2.2 NOM Approval

The NOM uses various tools including CMT and related LPOM calculations to review the proposed experiment and grants final approval via AppMan. The NOM will review all details regarding the component experiments, but need only provide approval for the parent.

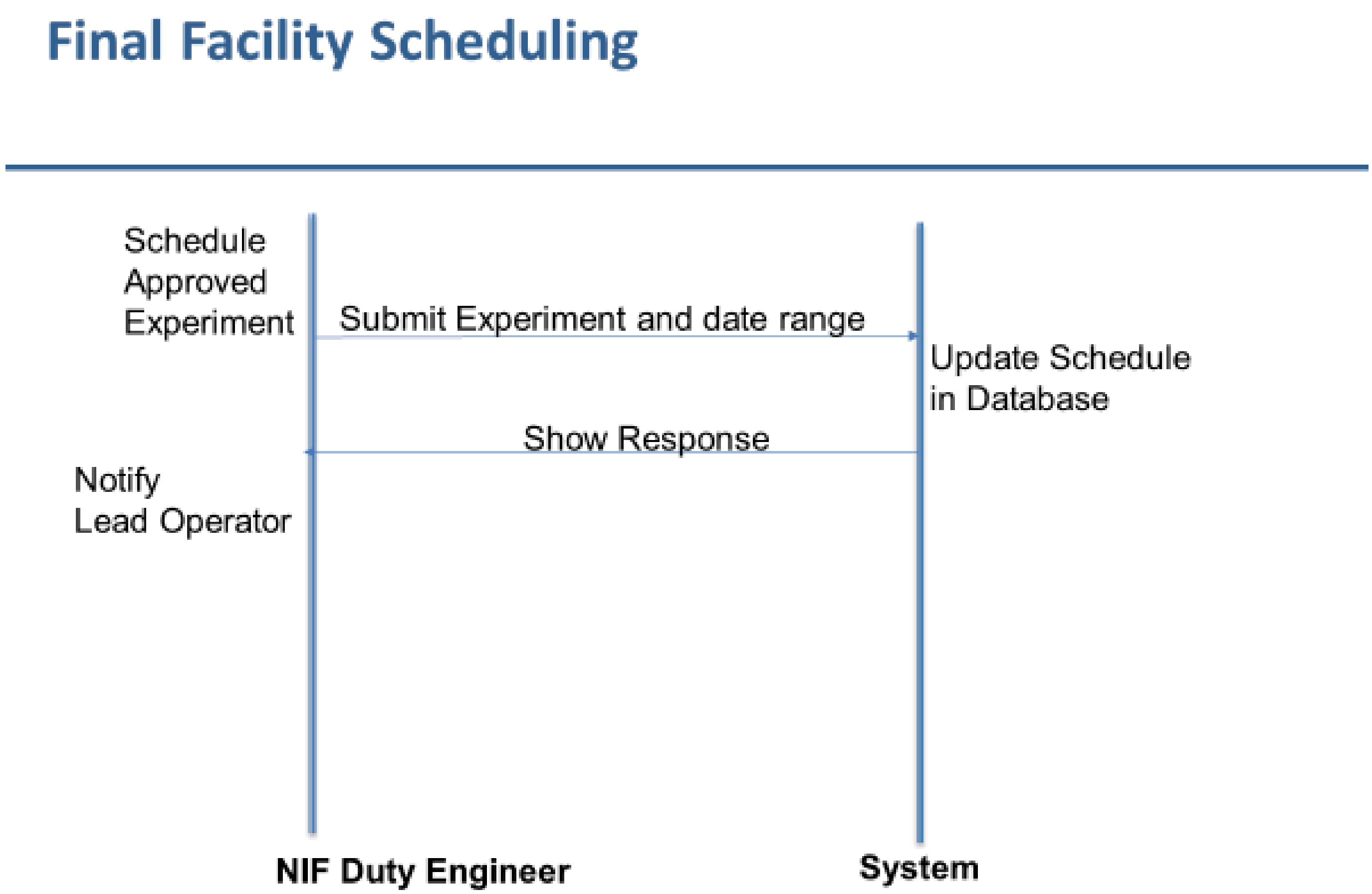
4.1.2.2.1 NIF Operations Manager Approval Sequence Diagram



4.1.2.3 Final Facility Scheduling

Once fully approved, the Duty Engineer schedules a single approved experiment for a specific date range. This action makes the experiment available on the "pull down" selection of the ICCS Shot Director Graphical User Interface.

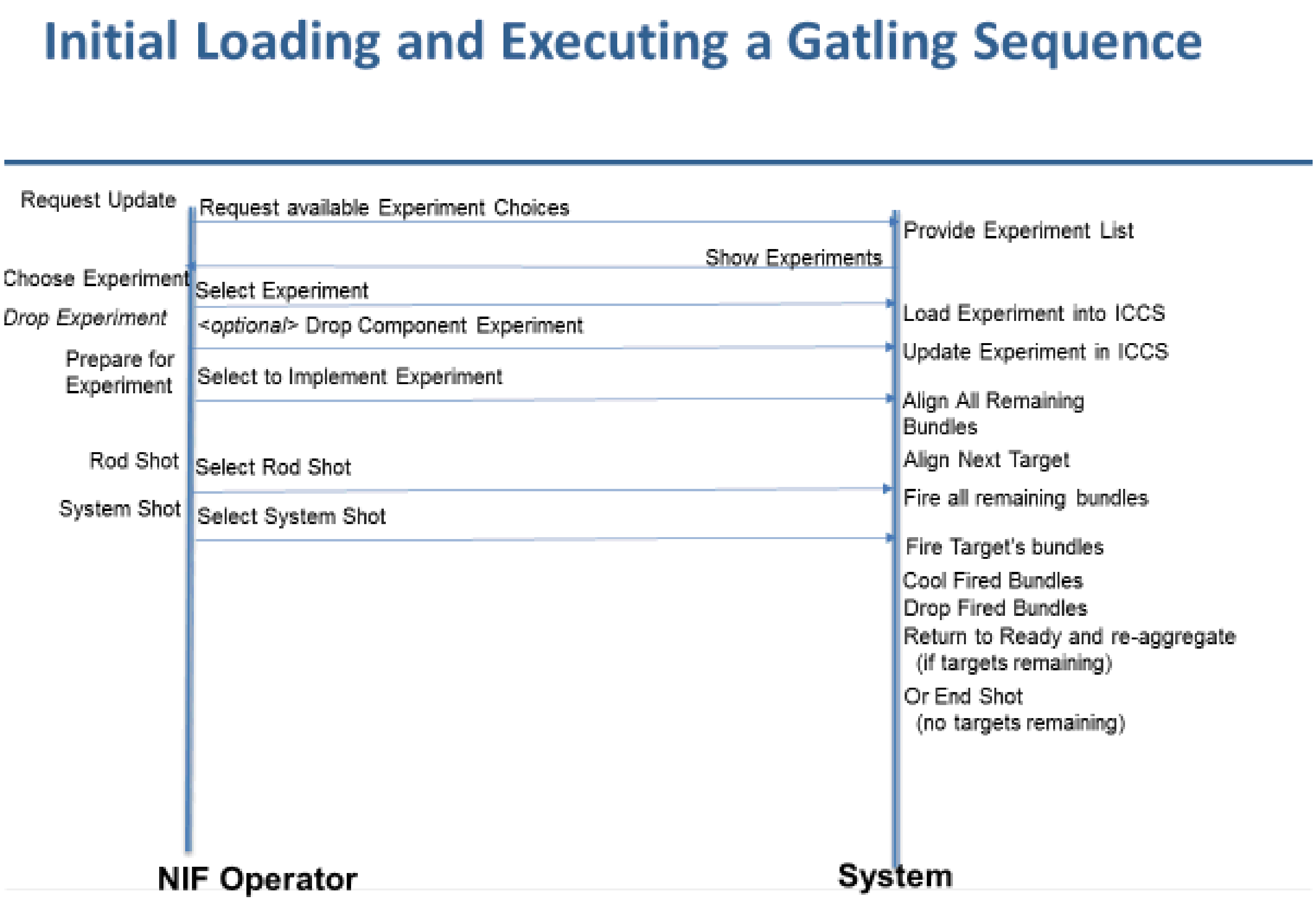
4.1.2.3.1 Final Facility Scheduling Sequence Diagram



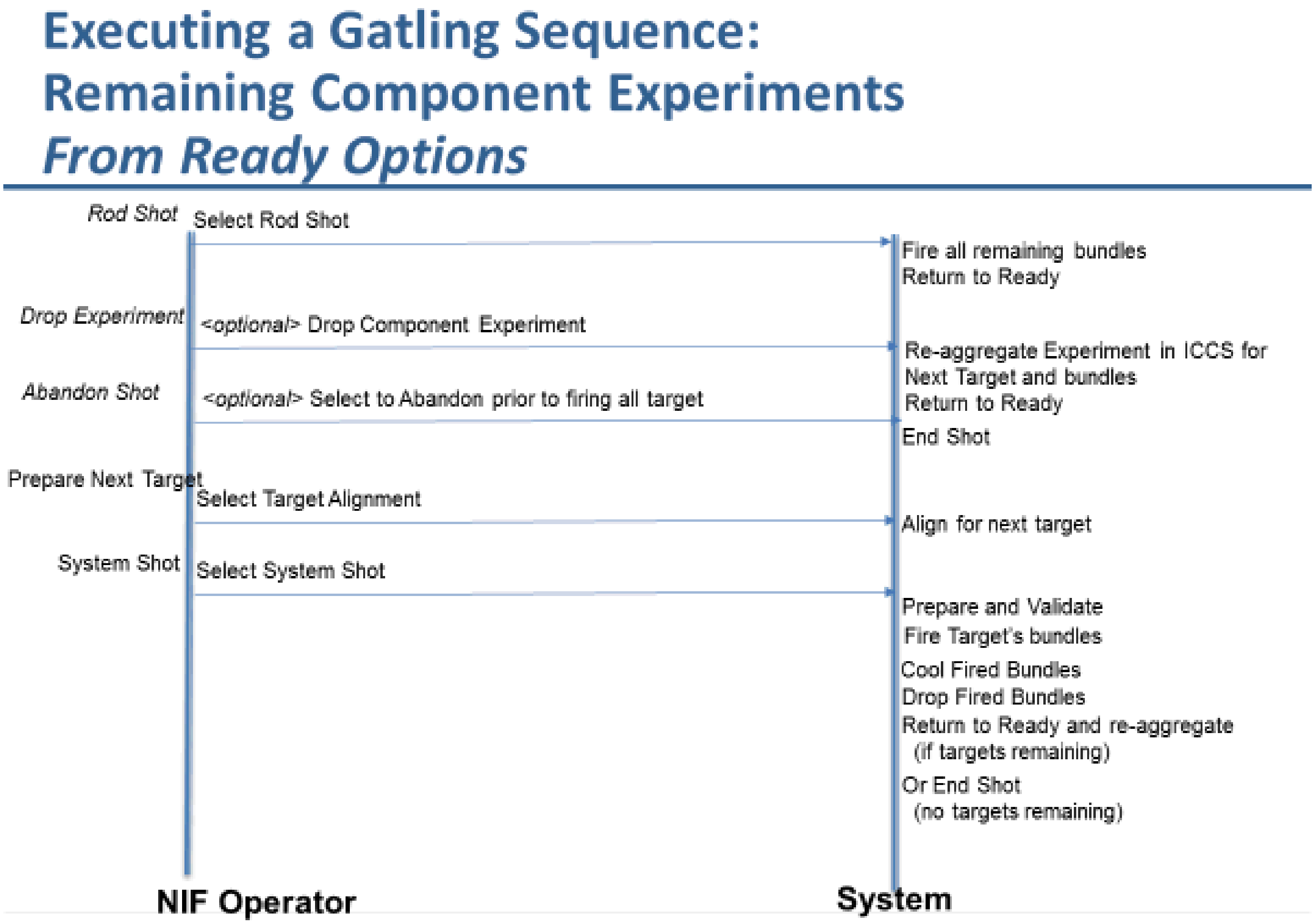
4.1.3 Execution (T -4 - T-10 Hours)

The NIF Operator (Lead Operator) will load the parent Gatling experiment. All the normal toolsets available to a standard experiment are available to this aggregated experiment. This experiment defines up to four component experiments and their defined order of execution. Shot Automation aids the NIF Operator in preparing for, executing, and adjusting a series of Gatling Shots by using and reaggregating the included component experiment information and its defined ordering.

4.1.3.1 Executing a Gating Shot Sequence: Initial Loading and executing Sequence Diagram



4.1.3.2 Executing a Gating Shot Sequence: Executing Remaining Targets Sequence Diagram



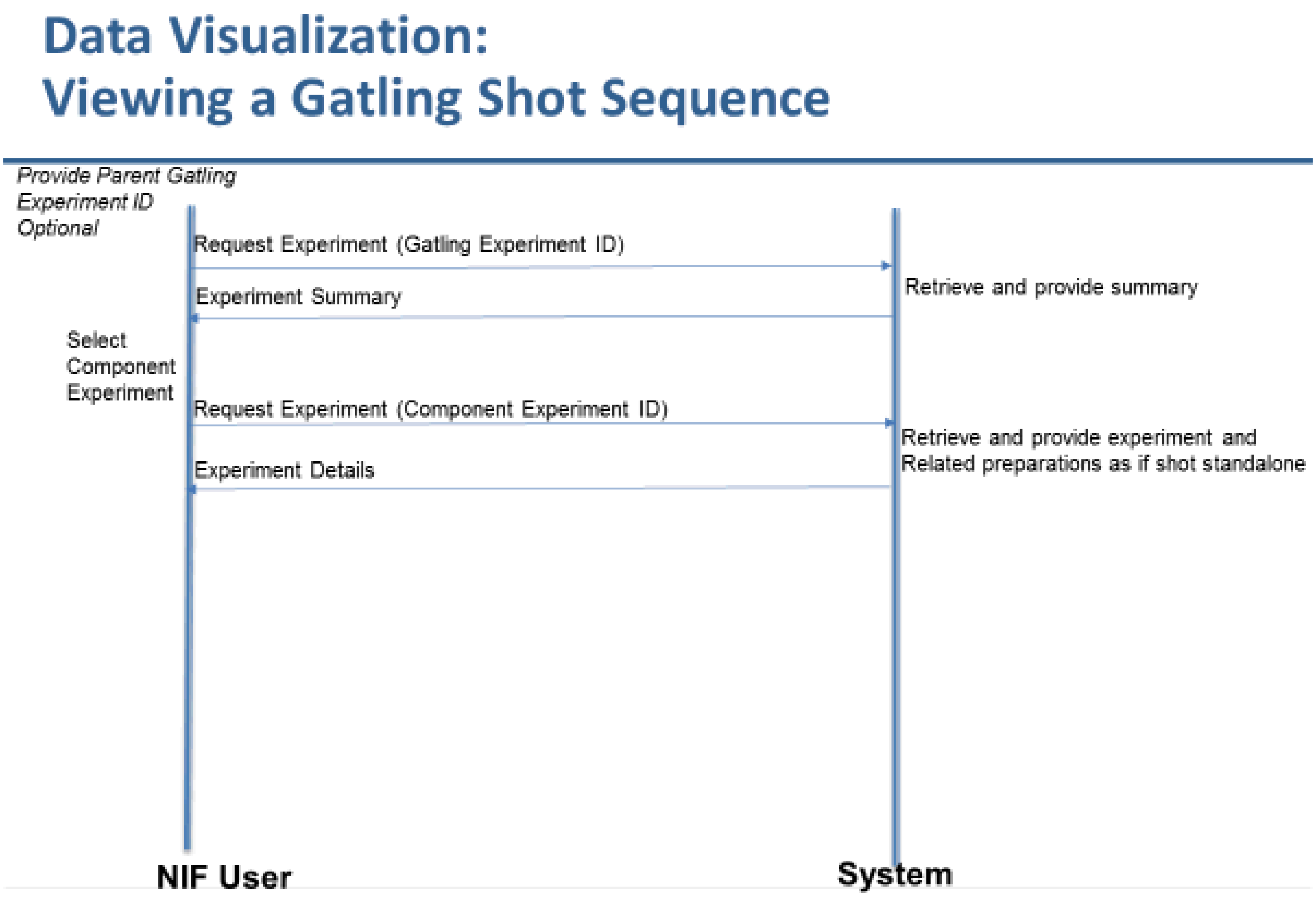
4.1.4 Data Visualization

4.1.4.1 Viewing a Gatling shot

Post shot, the NIF User may access the data associated with this Gatling Shot Sequence using Data Visualization. It is important here for the NIF User to be able to view the results of a component experiment as if it had been executed as standalone.

As stated above, the use of up to four component experiments also aids with relating "as fired" outcomes to specific experiments when archiving. Any component experiment may be reviewed as if fired standalone.

4.1.4.1.1 Viewing a Gatling shot Sequence Diagram



Section 5

Trade Study: Plausible alternatives and rationale

5.1 Alternative Implementation Concepts

NIF is originally designed to be single experiment based. By walking through operational scenarios and discussions with NIF Users through NIF Operations, it is clear that our current toolsets and procedures expect a defined experiment. It is also clear that at present a single experiment cannot define separate target and diagnostic needs of four different target system shots. From defining a Flip ID for an upcoming shot to defining and reviewing the shot itself, to reviewing its outcome, hundreds of person hours have been invested in providing the toolsets to manage individual experiments.

The sub functions of NIF are also clear: Shot Planning, Shot Execution and Data Visualization. The reference architecture tells us which of the existing systems support each of these sub functions. Here we use the Pugh Matrix to look at the comparable approaches for achieving the new capabilities allocated to each sub function.

5.1.1 Shot planner

During Shot Planning we need to easily recognize upcoming related Gatling experiments to aid in scheduling. Also, the NIF User needs to be able to enter multiple related Gatling experiments. While doing so, the average user should not be misled or confused by options or procedures they do not need. Rather than complicate Shot Planning for current users and/or risk breaking existing functionality of Shot Planner, we will incorporate minor interface additions and add a new System “**Experiment Editor**” to allow NIF Users concentrating on Gatling style shots to “bulk input” Gatling Shot proposals.

Pugh Concept Selection Matrix
Gatling Shot Planner

KPPs	1: Change Shot Planner allowing gatling input	2: Add Experiment Editor isolating gatling input to minor interface change on Shot Planner
1 Shot Rate	S	S
2 Full System	S	S
3 Generate Data	S	S
4 Don't break existing	-	+
5 Minimize User Impact	-	+
ε+	0	2
ε-	2	0
Satisfactory	3	3

5.1.2 Configuration Management Tool

During Shot Planning we need to easily input all the detail related to target and diagnostic information in a way which is familiar to a NIF User. Here again, the average 97% user should not be misled or confused by options or procedures they do not need. Rather than complicate CMT we will incorporate minor interface additions and add a new System "Gatling Shot Composer" to allow NIF Users concentrating on Gatling style shots to validate, aggregate, and order a Gatling Sequence providing all appropriate linking information necessary to fully automate execution of the parent experiment and all targets. This approach leverages usage of all existing experiment toolsets and procedures minimize user impact and training. It should also be mentioned that this decision aids all of our Data Visualization because using the Gatling Sequence Parent and child relationships, Data Visualization may present each component experiment as if they had been executed standalone. Placing the new Gatling behaviors into a new small application isolates the risk of breaking existing CMT functionality and workflow.

Pugh Concept Selection Matrix
Gatling Configuration Management Tool

KPPs	1: Change CMT to allow user to input all data related to a Gatling Sequence	2: Add Gatling Shot Composer to merge (re-aggregate) component experiments
1 Shot Rate	S	S
2 Full System	S	S
3 Generate Data	S	S
4 Don't break existing	-	+
5 Minimize User Impact	-	+
ε+	0	2
ε-	2	0
Satisfactory	3	3

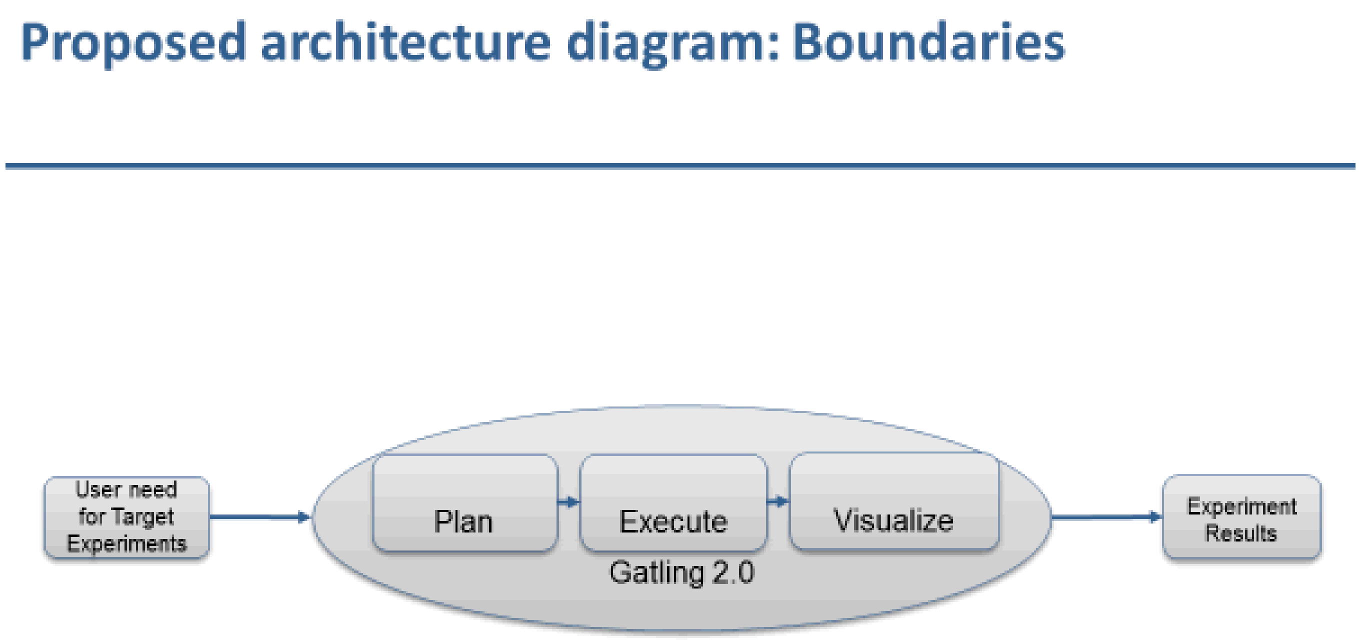
Section 6

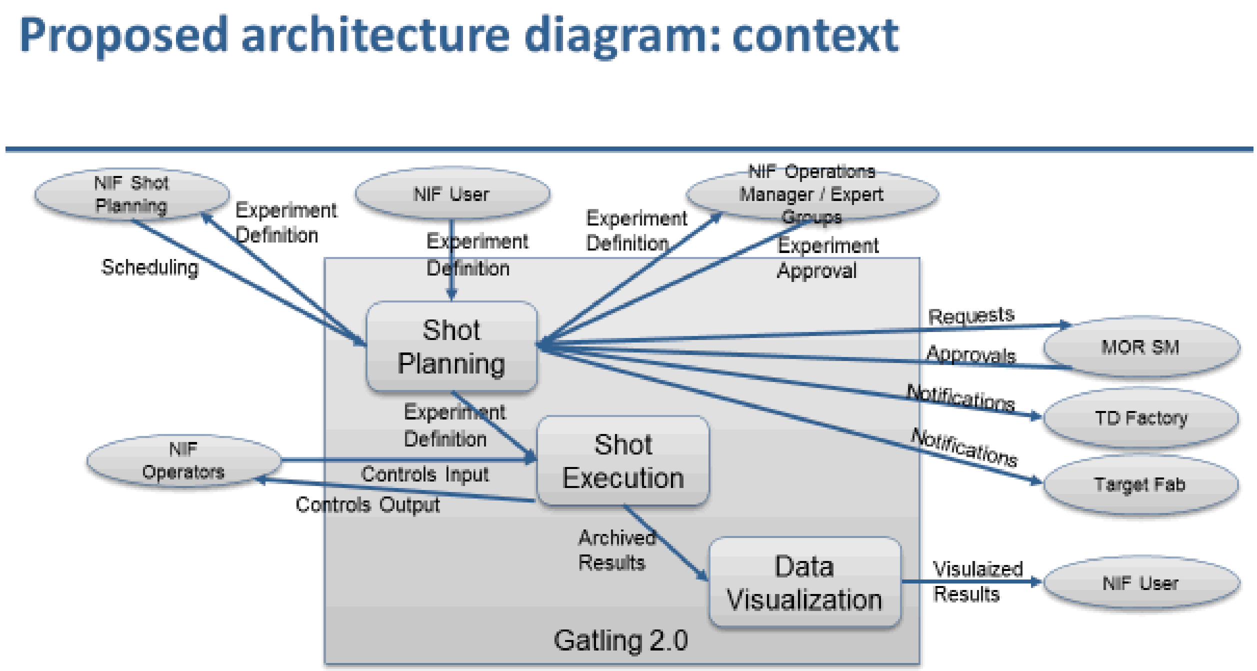
Use Cases

6.1 Proposed Architecture changes

6.1.1 Overview

At the top level, it remains the same.

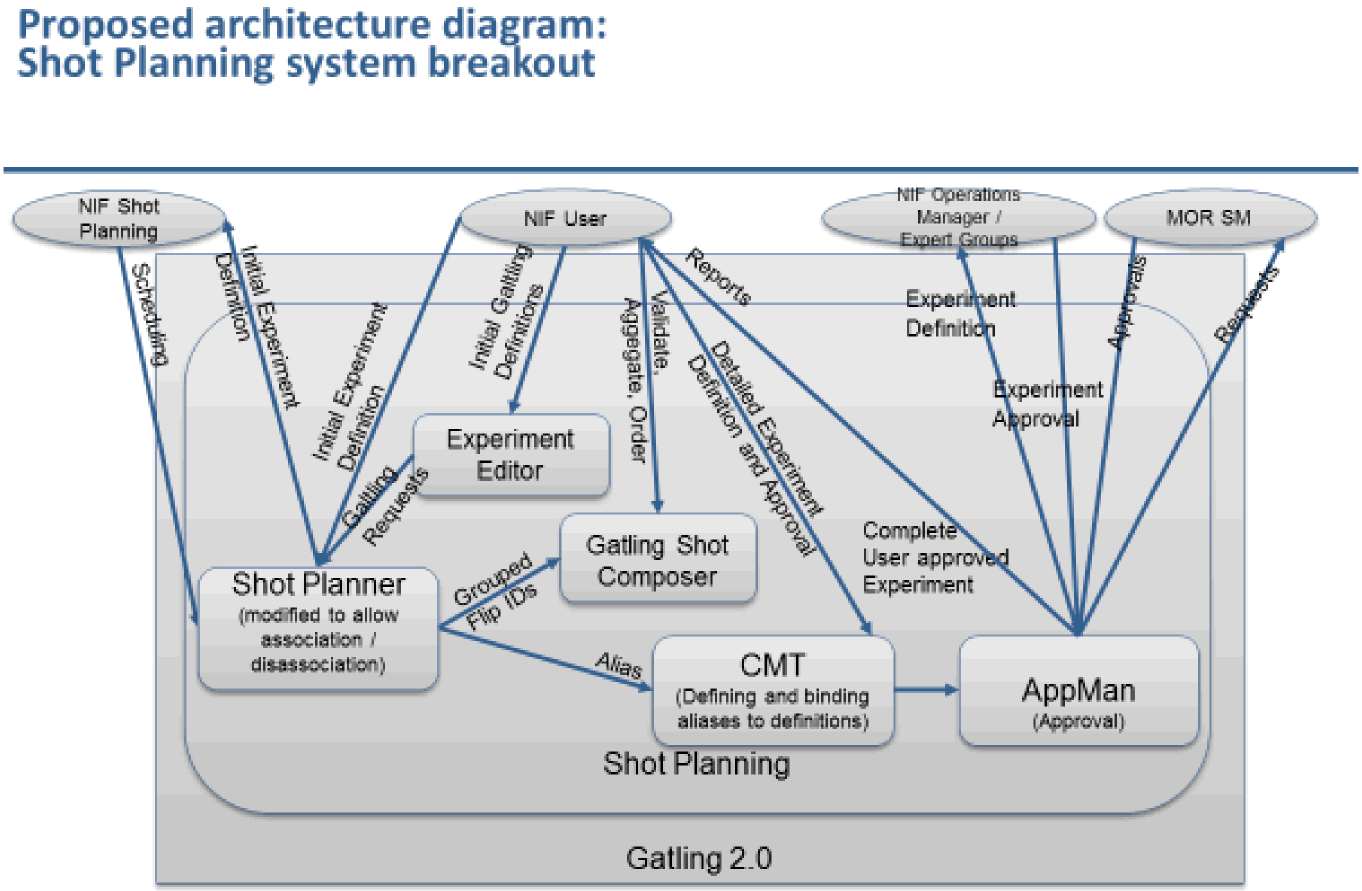




6.1.1.1 The bulk of Architecture changes are in Shot Planning

The majority of architectural changes are in Shot Planning: bulk input of Gatling Flip IDs via Experiment Editor, generating and viewing of Gatling Flip IDs via Shot Planner, validation and generation of loadable "fifth" experiment via Gatling Shot Composer, and changes to approvals in AppMan.

6.1.1.2 Shot PlanningArchitecture



6.1.1.3 Shot Execution changes center on interface and behavior changes

Internal changes here include: loading of parent Gatling, access to components for reaggregation, shot setup changes, shot execution changes including archiving of Parent Experiment ID, Shot Lifecycle ID, Component Experiment ID, and active Shot ID.

6.1.1.4 Data Visualization changes center on interface and behavior changes

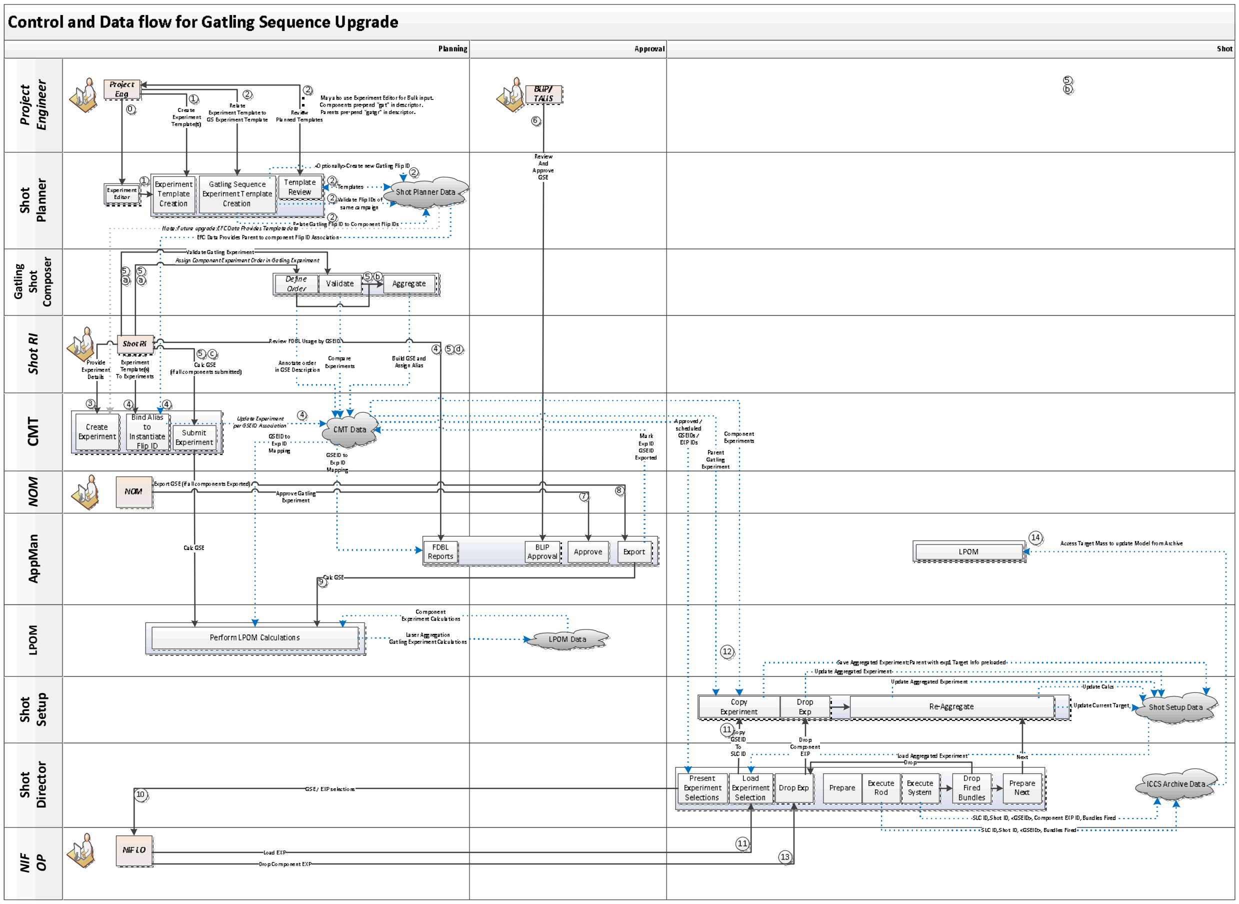
Changes here include: use of Gatling Sequence information to browse and display shot results as if standalone.

6.2 Proposed use case overview

The ***NIF User*** defines a set of "Flip IDs" related to a Gatling Experiment (up to four) and their parent experiment, either by bulk input in the **Experiment Editor** (new) or one at a time using the **Shot Planner** (modified). Using the **Shot Planner** (modified), the ***NIF Scheduler*** can see which Flip IDs are associated to a Gatling Experiment and considers this in planning facility time. The

NIF User goes on to further define the details associated with each experiment using **CMT**. When the details of all component experiments of a parent Gatling Experiment have been provided and submitted, the ***NIF User*** may validate, aggregate, and order the parent Gatling Experiment using the **Gatling Shot Composer (GSC)** (new). This assures that the component experiments do not conflict in areas such as cluster use. The **GSC** creates a “fifth” experiment which will allow laser preparation for all four target shots. With a valid fifth experiment submitted the ***NIF Expert Group*** may review all details of the Gatling Experiment and approve their specific areas of expertise using **AppMan** (modified). **AppMan** will manage distribution of approvals made at the parent level, as well as roll up of component approvals. With all ***NIF Expert Group*** approvals in place, the ***NOM*** may approve the parent Gatling Experiment for execution in the facility using **AppMan**. Once approved, the ***NIF Operator*** (Duty Engineer) will perform final scheduling via the database to allow loading of the parent Gatling Experiment in **Shot Execution** (ICCS modified). At this point the Gatling Experiment will be available on the NIF Shot Director Graphical User Interface for selection. The ***NIF Operator*** (Lead Operator) will select to load the experiment in **ICCS** and begin preparation full system laser alignments and target alignment of the first target in the sequence. Using information provide by **Gatling Shot Composer**, **ICCS** will align and fire each target one at a time under separate System Shot IDs allowing easy review later in **Data Visualization** (modified). Between each shot **ICCS** will work in conjunction with **Shot Setup** to “re-aggerate” the details of a component experiment into the parent experiment. **ICCS** will then allow the NIF Operator to automatically align and prepare for the next system shot based on the order embedded in the parent Gatling experiment by the **GSC**. Once all targets have been shot, the **ICCS** Shot Automation software will end the Shot Lifecycle automatically. **LPOM** will use the information associated in the archive with each target shot to update the Target Mass information. The ***NIF User*** may then use **Data Visualization** to review the results of each experiment either by accessing them from the parent Experiment ID, Shot Lifecycle ID, or component Experiment ID. If accessed by component Experiment ID, **Data Visualization** will present the shot outcome as if the experiment had been fired as standalone, including preparation and rod shot information.

ConOps overview slide used in Stakeholder discussions



6.2.1 Individual Use Cases and Sequence Diagrams

6.2.1.1 Shot Planning addition of bulk input details

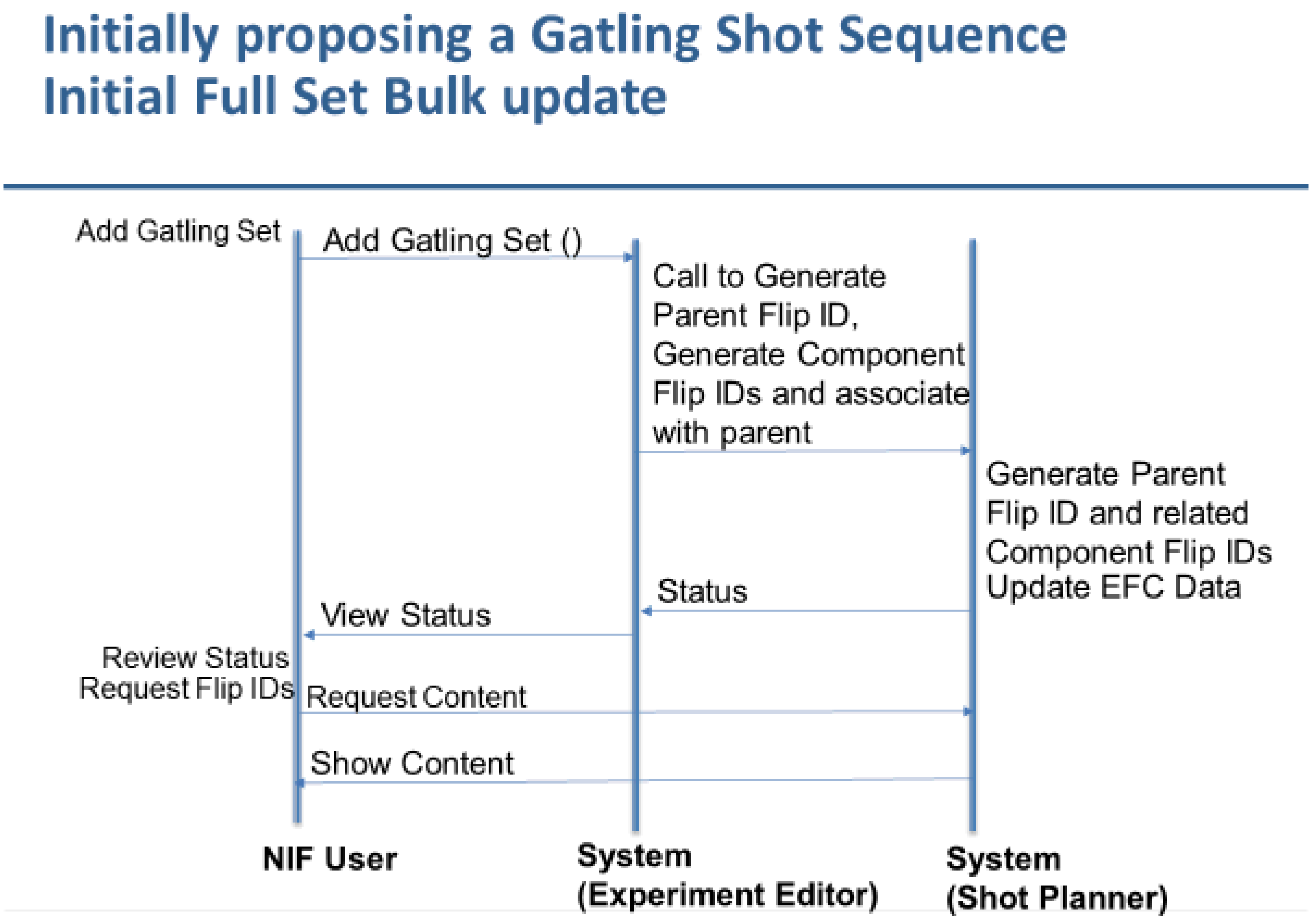
Here the **NIF User** uses the **Experiment Editor** to bulk input experiments and, in doing so, relate them to one another when intended to be performed in a Gatling context. Using additional interfaces provided by Shot Planner, the experiment editor will generate unofficial Flip IDS by prepending “gat” in the descriptor of any unofficial Flip ID intended to be performed within a Gatling Experiment. The “parent” or Gatling Experiment containing it would be prepended with “gatgr.” This distinction via the descriptor will separate alias sequencing for standard, component Gatling and group Gatling Flip IDs. You will find that this modification to the descriptor field is central and, in fact, simplifies the Con Ops regarding individual experiment approval. Note that each sibling knows its siblings and the parent knows all of its children.

In Experiment Editor: unofficial Flip IDs

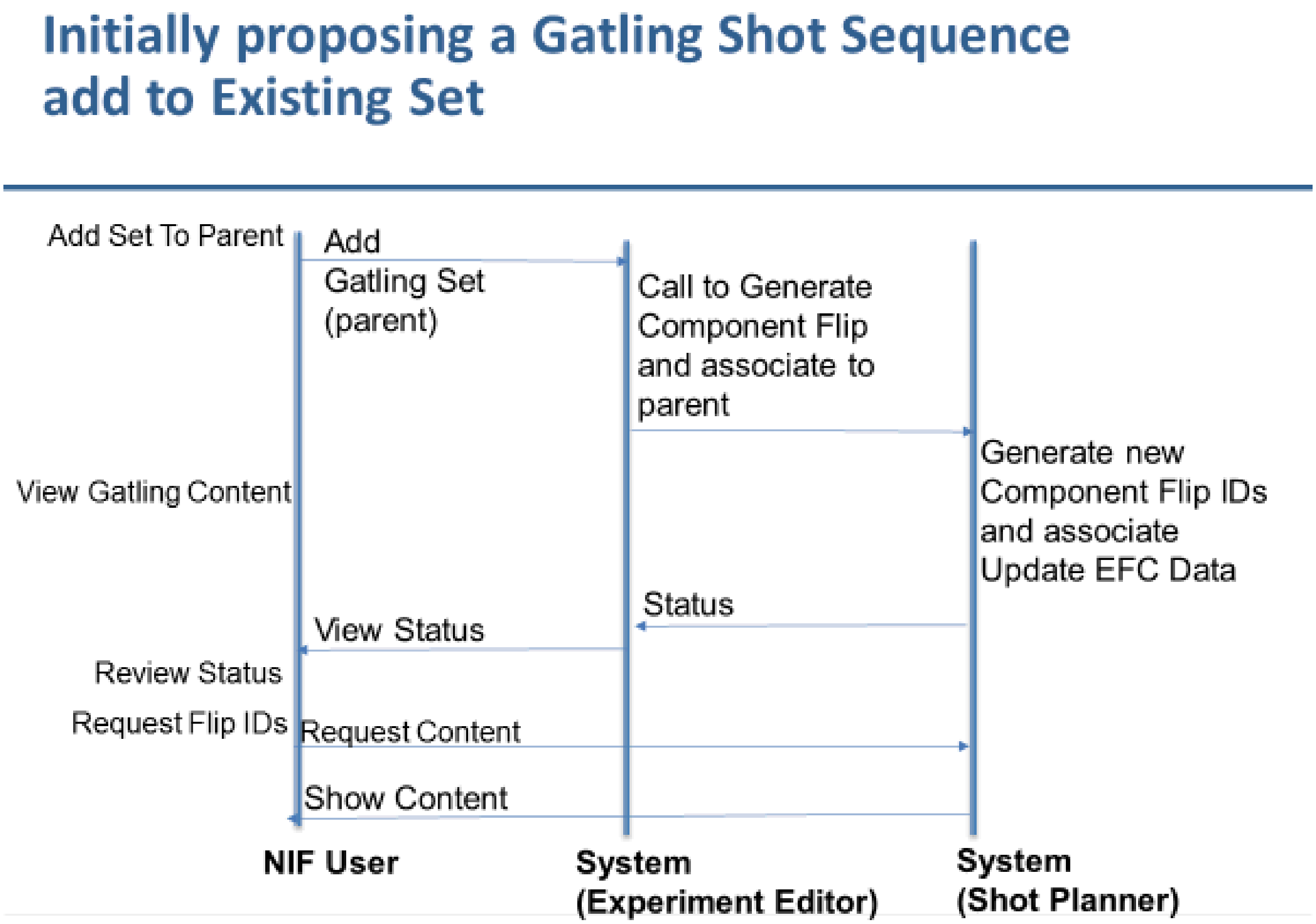
Standard:	Fa_Ncap_DISKBL_PQ_E01
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_E01
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_E02
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_E03
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_E04
Gatling Group:	Fa_Ncap_DISKBL_gatgrPQ_E01

Provides:
New Shot Experiment Request GUI to user capable of associating experiments to Gatling execution.
Inputs:
from user through GUI
Outputs:
Makes calls to **Shot Planner,** creating and associating Flip IDs to a parent Gatling Flip ID
Resource Needed:
Shot Planner interfaces allowing association/disassociation

6.2.1.2 Shot Planning addition of full set bulk input Sequence Diagram



6.2.1.3 Shot Planning addition of partial set bulk input Sequence Diagram



6.2.1.4 Shot Planning single initial component input details

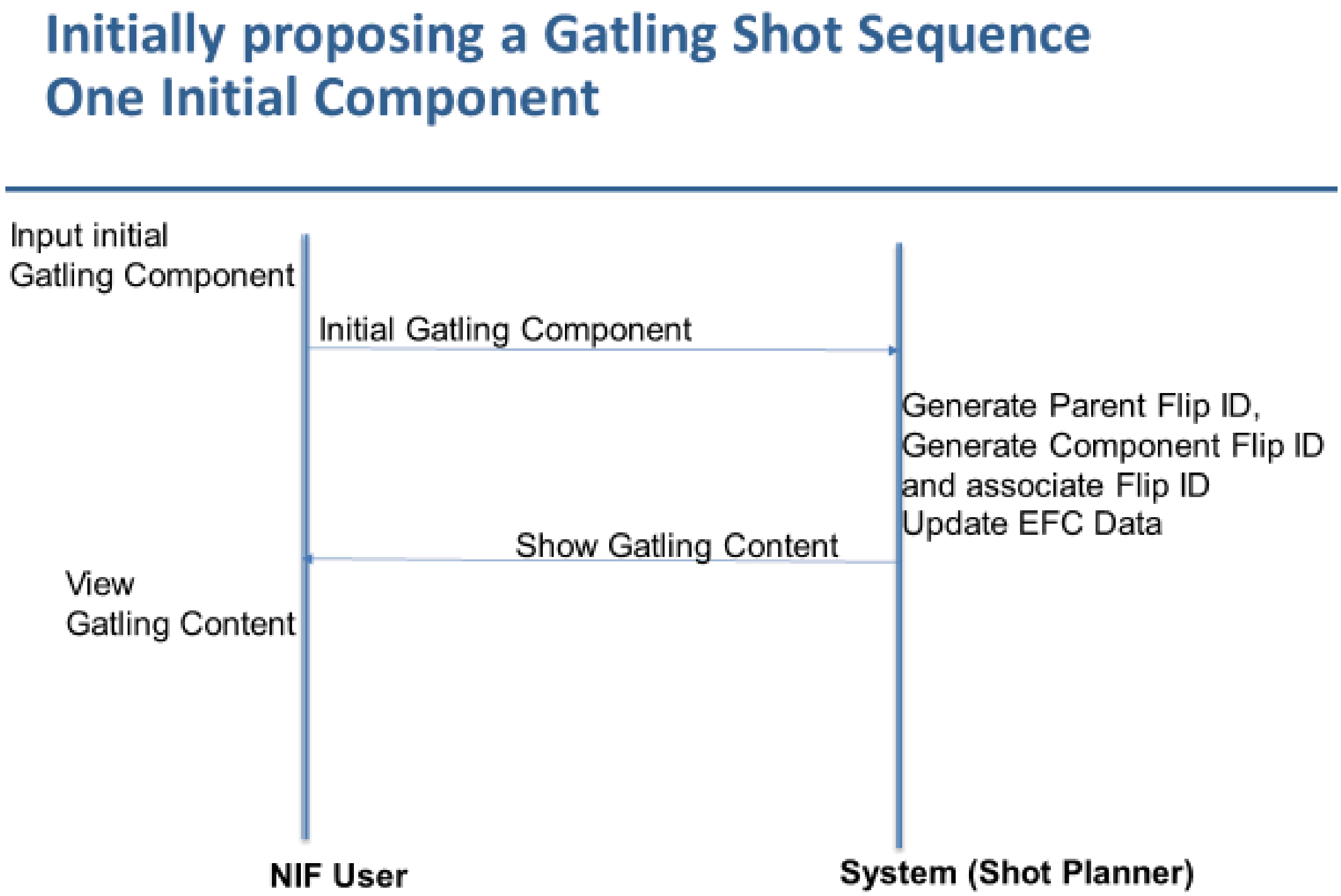
Here either the NIF User or Experiment Editor may call on the **Shot Planner** which will be modified to support entering and relating Gatling Experiments and at-a-glance review. It should be noted that the relations defined in the **Experiment Editor** will be leveraged if provided. If inputting by hand directly into **Shot Planner**, we will prepend “gat” in the descriptor of any official Flip ID intended to be performed within a Gatling Experiment. The “parent” or Gatling Experiment containing it would be prepended with “gatgr”. Note that each sibling knows its siblings and the parent knows all of its children. Note that hovering on a “gat” experiment will “popup” a list of its siblings. All component experiments in a parent are from the same Campaign. Shot Planner will be modified such that a new association, re-association or dis-association of any component experiment causes invalidation of the parent Gatling Sequence Experiment.

Provides: Gatling planning input tools to user including at a glance and float over relationships

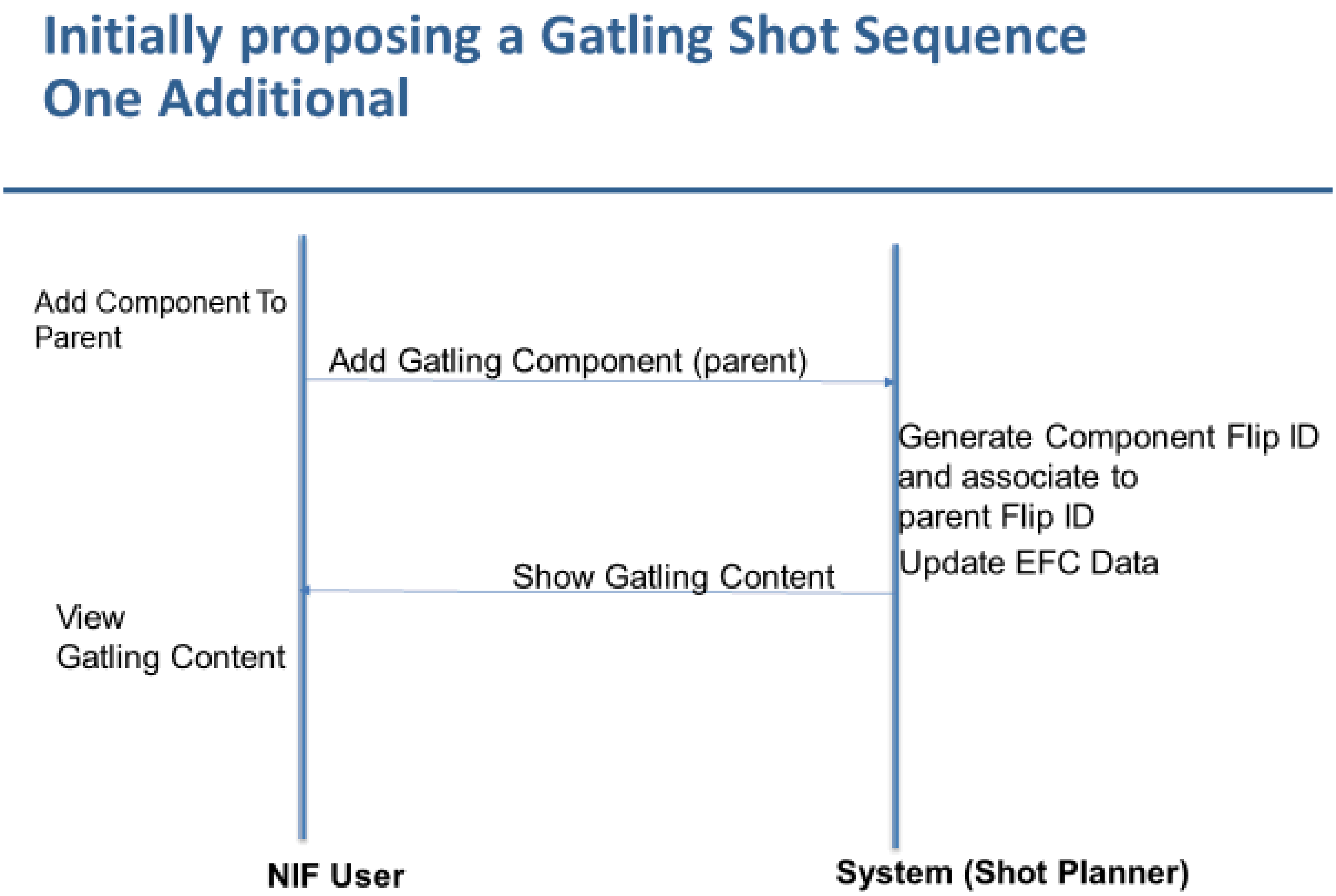
Inputs: Calls from Experiment Editor creating and associating Flip IDs to Gatling or user doing so directly

Outputs: Place holder parent Flip ID and associated Flip IDs via modified EFC Data used by Gatling Shot Composer

6.2.1.5 Shot Planning single initial component input Sequence Diagram



6.2.1.6 Shot Planning additional component input Sequence Diagram



6.2.2 Shot Planning Initial Scheduling

6.2.2.1 Shot Planner scheduling

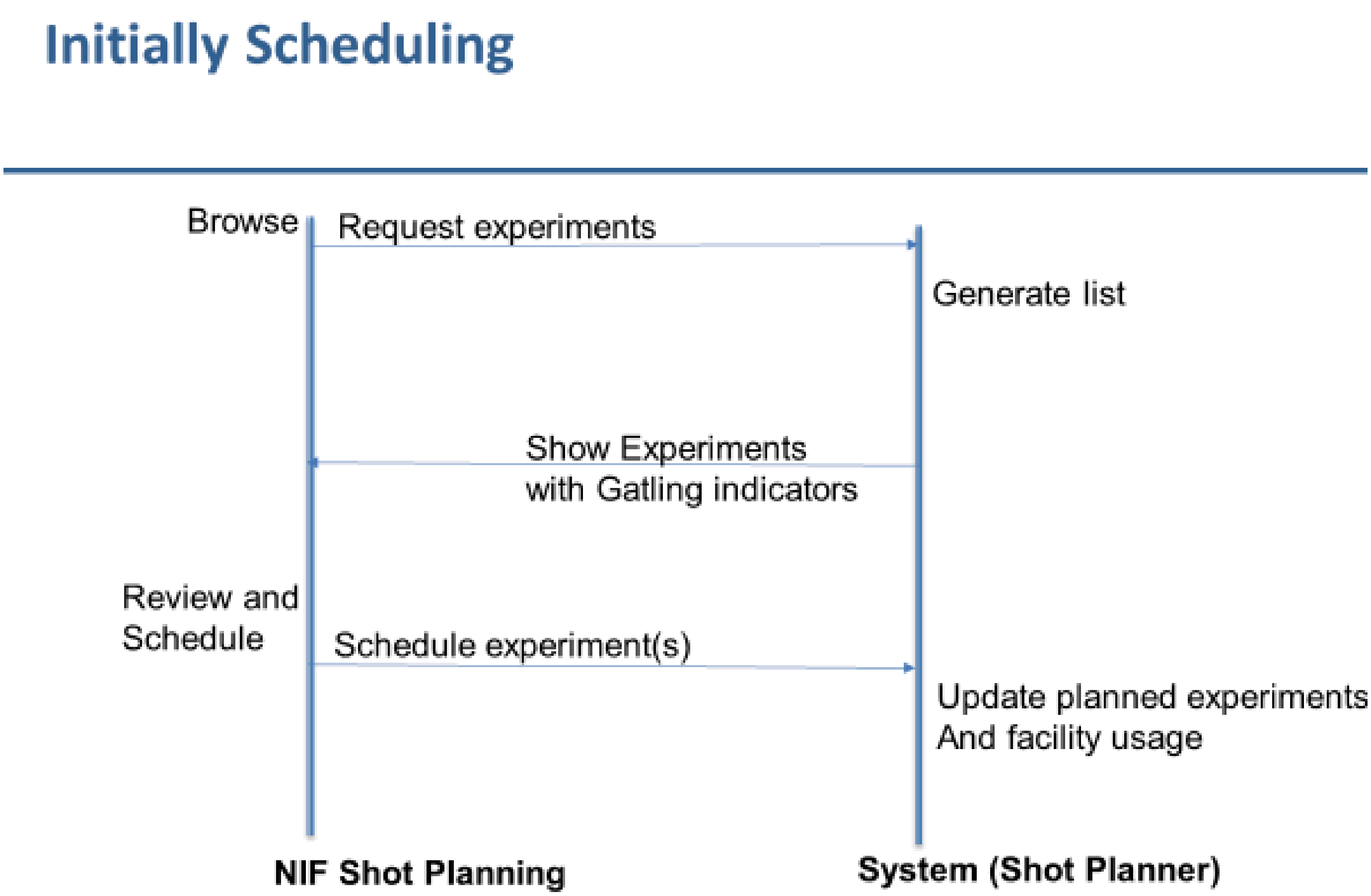
Here either the NIF Scheduler will review upcoming experiments and, via their Flip IDs, quickly relate Gatling shots to one another. Using this method the NIF Scheduler may allocate facility time relevant to the approach being taken. In addition, floating over an experiment will display any Gatling relationships if so defined.

Provides: Gatling planning tools to user including at a glance and float over relationships

Inputs: User

Outputs: Place holder parent Flip ID and associated Flip IDs via modified EFC Data used by Gatling Shot Composer

6.2.2.2 Shot Planner scheduling Sequence Diagram



6.2.2.3 *Further Defining Experiment Content*

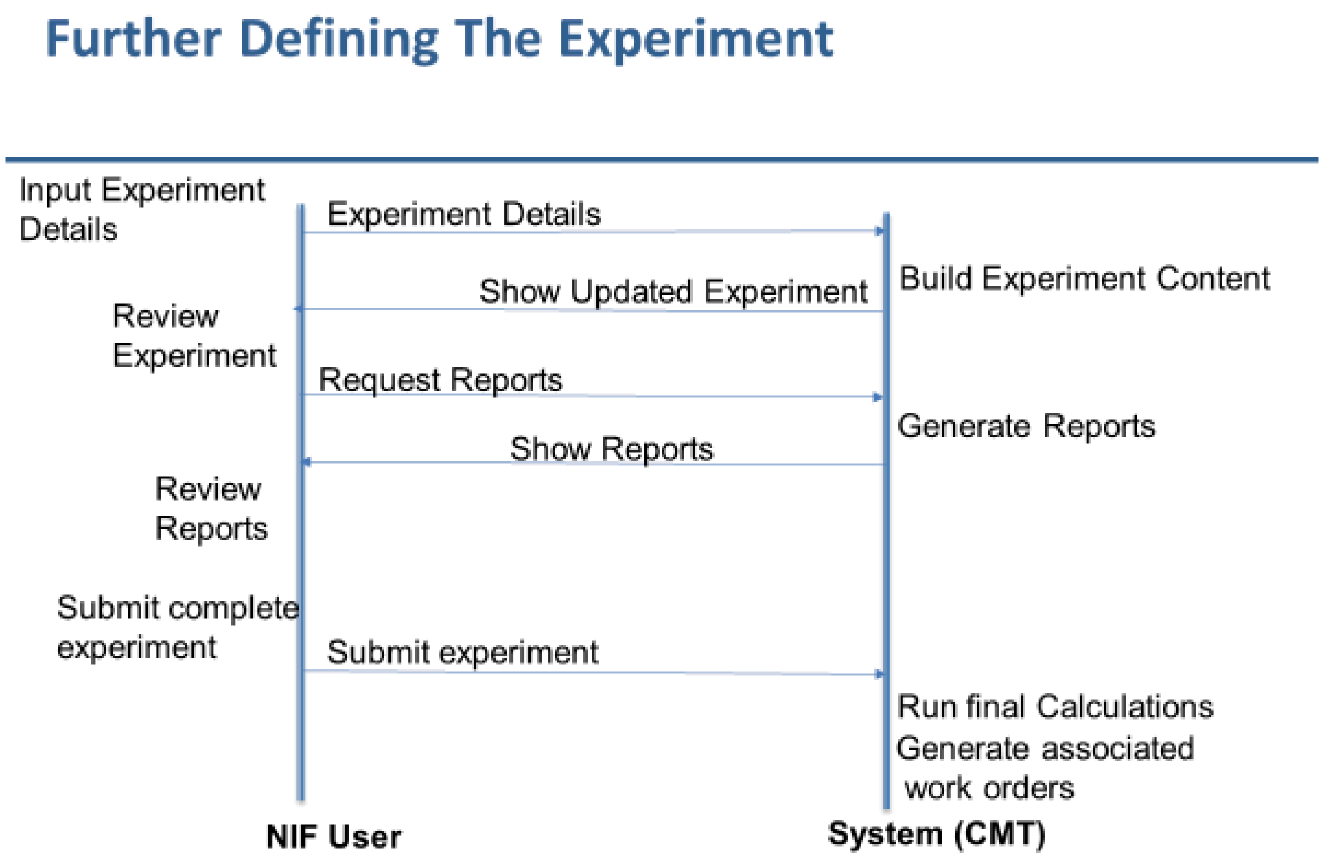
The NIF User already uses **CMT** to associate aliases to experiments to instantiate Flip IDs; here it is modified to associate component experiments with their parent experiment. CMT will be modified to leverage the cap information, indicating components and order, of the parent Gatling Experiment. CMT will support pre-export calculation of the Gatling Experiment in LPOM (note no change). CMT will be modified to provide Gatling Experiment Overview information containing Parent Experiment and components. CMT will be modified to detect changes to Gatling dependent settings within a component experiment and invalidate the validation and aggregation done by the GSC. The experiment XML schema will be modified to add the “cap data” populated by the GSC and used both here and in AppMan. CMT should not submit a Gatling Parent to LPOM unless the Gatling Valid is true. CMT will be modified such that a New association, reassociation or disassociation of any component experiment causes invalidation of the parent Gatling Sequence Experiment.

In CMT:

Standard:	Fa_Ncap_DISKBL_PQ_S01
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_S01
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_S02
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_S03
Gatling Component:	Fa_Ncap_DISKBL_gatPQ_S04
Gatling Group:	Fa_Ncap_DISKBL_gatgrPQ_S01

Provides:
Gatling Experiment overview, association and disassociation of Flip IDs, modification of experiment details causing Gatling Experiment withdrawal and Gatling Valid reset to false.
Inputs:
Direct user input
Gatling Experiment cap data (Components and order),
Gatling Valid Flag
Outputs:
User Overviews, Withdrawals per change, Valid Gatling reset per change, XML Cap data describing Gatling Group relationships
Resource Needed:
Cap Information, Gatling Valid Flag

6.2.2.4 Further Defining Experiment Content Sequence Diagram



6.2.2.5 *Further Defining Experiment Order*

A standalone **GSC** will be created which will validate component experiment compatibility, define the order of their execution and aggregate/build a parent Gatling experiment from the components. To do this the GSCwill use "EFC Data" passed from **Shot Planner** describing relations between experiments.

Prior to aggregating a "fifth" experiment from four components this application must ***validate*** that the components are compatible.

Component Experiment compatibility is broken into three categories: Bundle, Diagnostic and Target.

Bundle:

- Use of any Quad or Bundle within a Cluster constitutes use of that Cluster and a Cluster may not be used in more than one of a Gatling Experiment's Component experiments.

Diagnostic:

- Diagnostic usage within a component experiment will continue to be validated at the individual experiment level.
- The Gatling Composer will validate that, per DIM, per component experiment, if the DIM is used the Filter and Snout are the same. Not being so would be an Error.
- The Gatling Composer will provide a difference report across up to four component experiments
  - This difference report will be diagnostic specific
    - See **Fishler/Kalantar** for detailed requirements. (**Gatling Shot Composer**)
- *Notes and impacts:*
  - As guidance, reviewers should consider only breaking sweep between 2 and 3 to maximize savings and attempt to minimize the number of reconfigurations
  - Diagnostics used in component experiments may differ experiment to experiment
    - This indicates that dry runs are informed by "next up" component experiment content updated via re-aggregation and will run between system shots
    - New Target Change Flex Update System to address, FCM prompt, reading of goals, XML update, Diagnostic Alignment, and parallel Dry Run / Target Alignment

Target:* TAS *Position:

- *Beam TP group configuration identical across all experiments*
  - *If an experiment uses a TP group for beams it is configured the same in any other component experiment in which it is used*
    - *5 values: X, Y, Z Upper, Z Lower, Z TAS*
  - *Target in last tp group and configuration may vary due to target metrology*
    - *Target only in last TP no Beams*

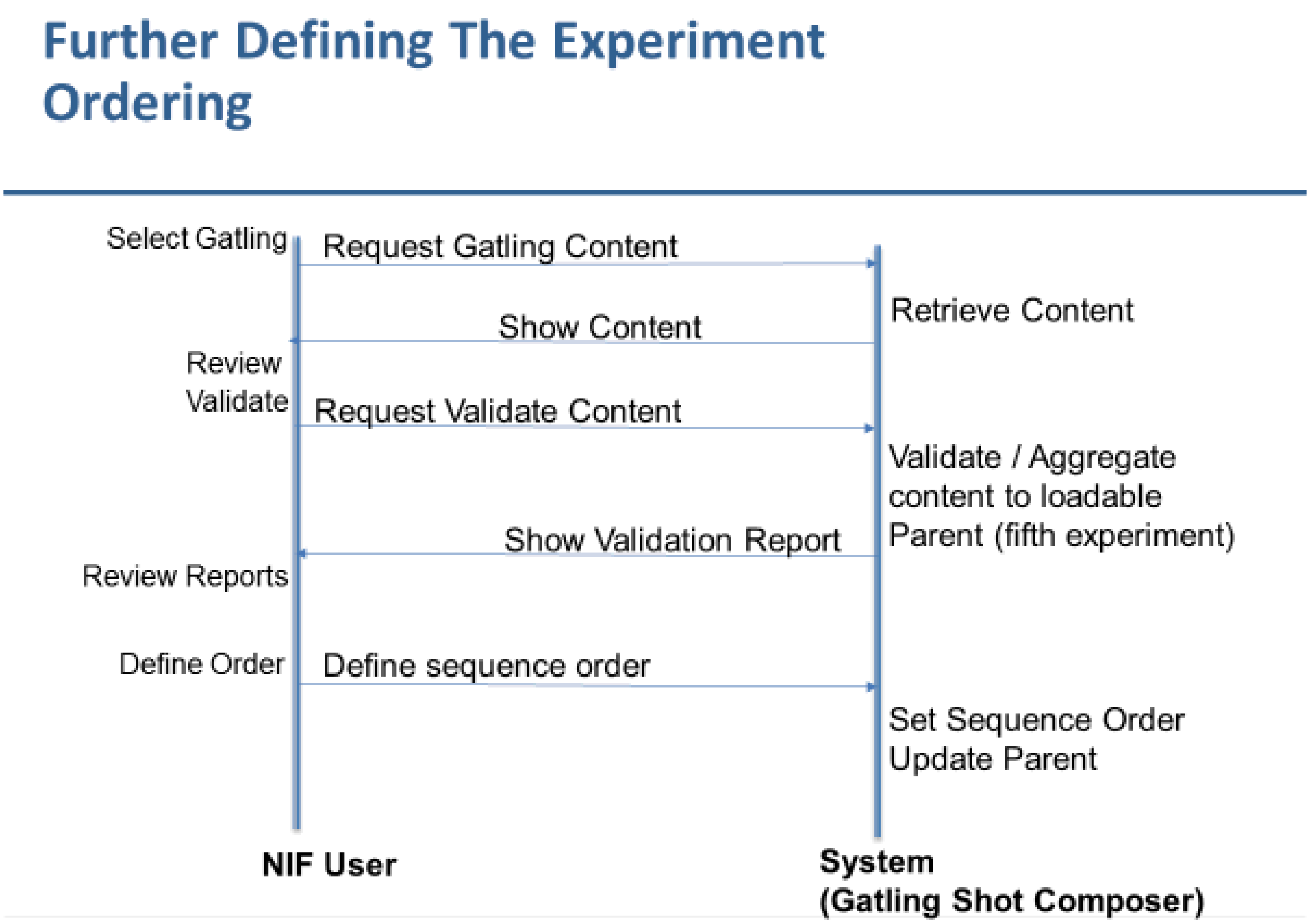
Gatling Shot Composer will provide clear and explicit feedback regarding TP use/value mismatch; however, an additional view across component experiments to aid RIs in aligning TP use would be extremely helpful. Once determined as valid, this application will be capable of aggregating Gatling component experiments into a parent Gatling Experiment suitable for laser setup, LPOM calculation. It should be noted that the parent Gatling Experiment is, in fact, the experiment that will be loaded to execute the sequence of component experiments. As such, it will also be used to plan and implement FDBL needs associated with all component experiments.

Using this application, a NIF User will define the order in which the component experiments will be executed. This cap information will be associated to the parent Gatling Experiment. All future experiments will have a newly created Associated Gatling Experiment ID field (GSEID). The parent experiment will have itself annotated in its Associated GSEID field. Non Gatling Experiments will have a null Associated GSEID field.

The GSC will set the CMT Shot Type in the parent to GATLING. Once validated, ordered and aggregated, the parent will be marked as Gatling Valid true.

Provides:
the ability to validate, aggregate and order the component experiments within a parent Gatling Experiment.
Inputs:
Place holder parent and associated Flip IDs via modified EFC Data
Outputs:
A *fifth*, aggregated experiment containing aggregated laser setup,
Cap information associated with the parent Gatling Experiment defining component experiments and their order,
an Associated Gatling Experiment ID on every experiment,
Gatling Valid Flag
Resource Needed:
updated EFC data defining Gatling flip ID associations

6.2.2.6 Further Defining Experiment Order Sequence Diagram

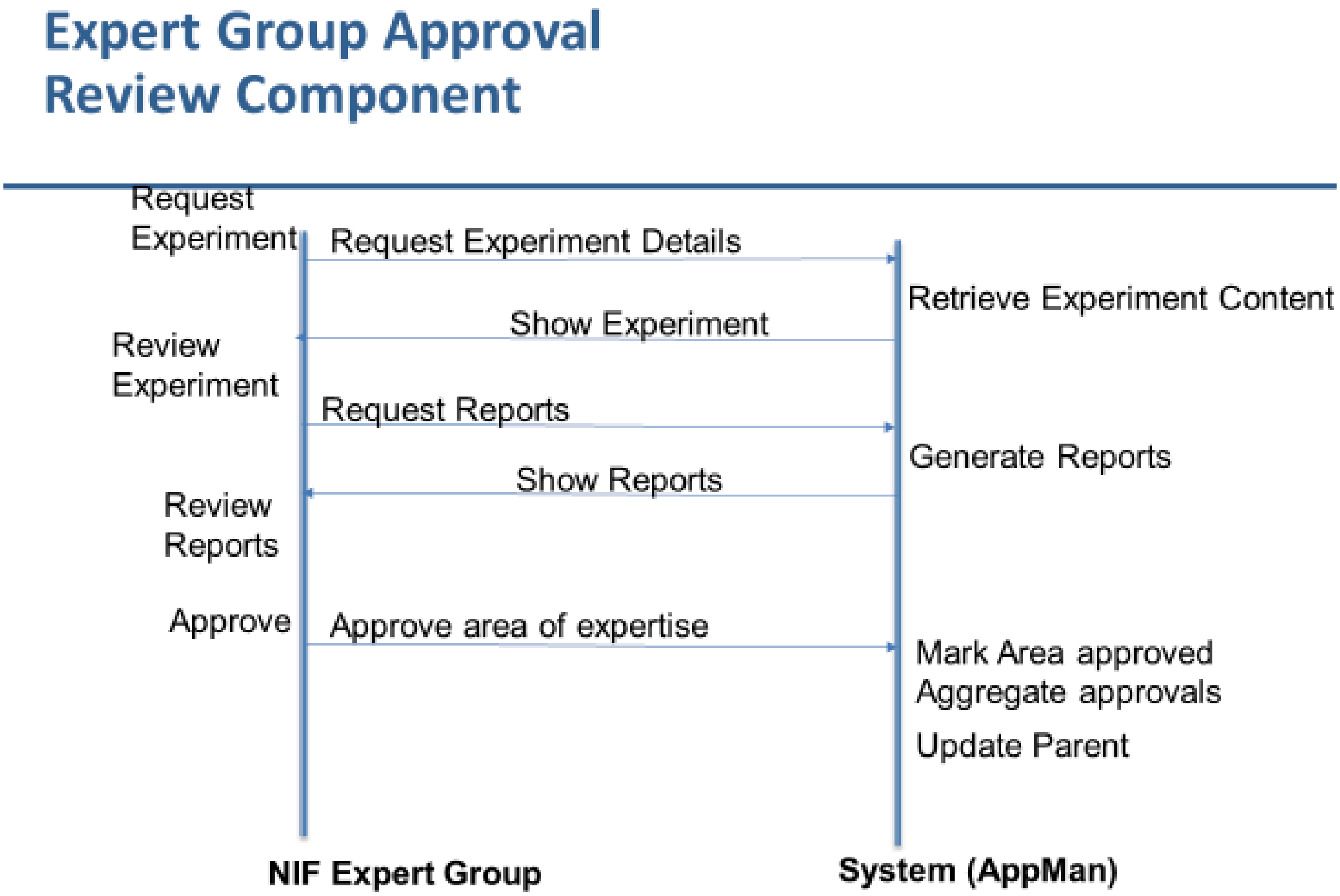


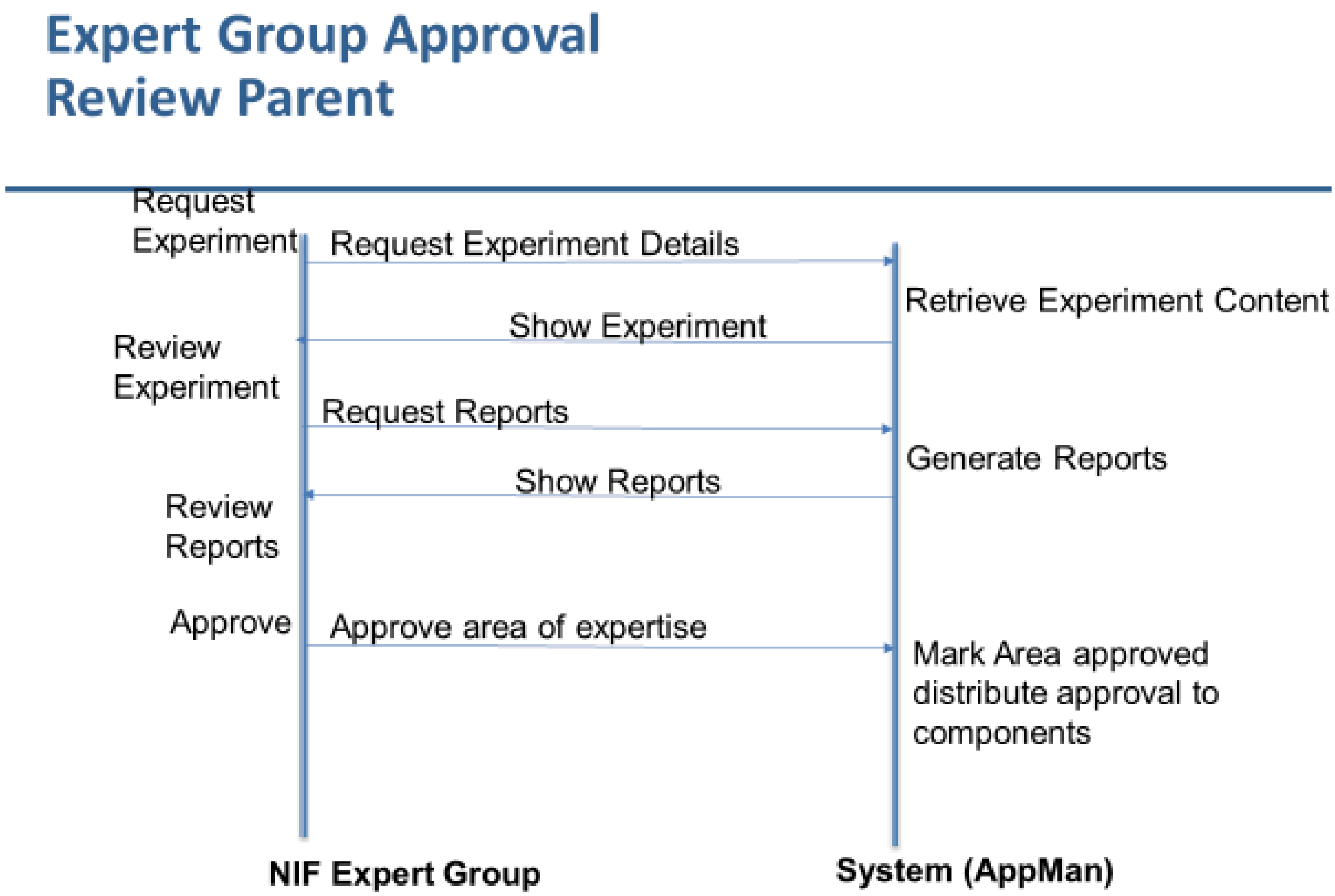
6.2.2.7 Expert Group Experiment Approval

AppMan will be modified to support review and approval of the parent Gatling Experiment. AppMan will be modified to consider component approval status when approving the Gatling Parent and provide warning/override capabilities where not all are approved. AppMan will be modified to not allow export of a component or parent experiment if the parent is not valid. A Gatling Sequence may not be exported until all of its component experiments have been exported. AppMan will be modified to support a diagnostic usage report using shot order in order to allow users to minimize the number of diagnostic reconfigurations. Blip will use review of the aggregated "fifth" experiment to review laser setup. Withdrawal of a component will initiate withdrawal of the exported parent or fail. Note also, here the information generated in the GSC regarding ordering will be pertinent to shot approval.

Provides:
Review and approval of Gatling Experiments
Inputs:
Direct user input,
Gatling Experiment cap data (Components and order),
Gatling Valid Flag
Outputs:
User Overviews,
Diagnostic usage difference reports,
Approvals, Warnings,
Calls CMT to export
Resource Needed:
Cap Information, Gatling Valid Flag

6.2.2.8 Expert Group Experiment Approval Sequence Diagrams

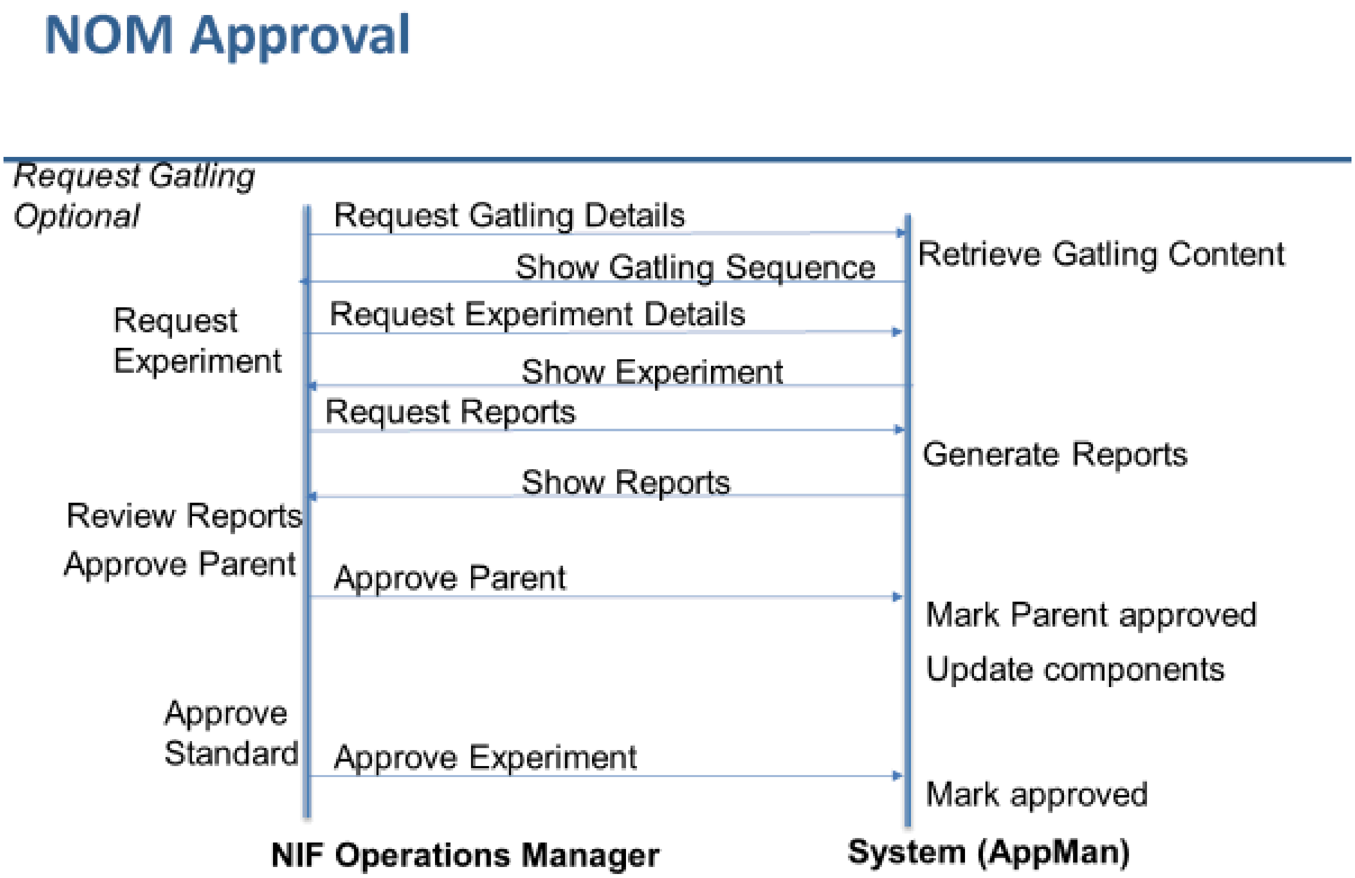




6.2.2.9 NOM Approval of Gatling Shot

As stated above, **AppMan** will be modified to support review and approval of the parent Gatling Experiment. Here the NOM will use AppMan to review and approve a Gatling shot sequence via browsing its contents from the parent and then approving from the parent. Also, as stated withdrawal or modification of any component experiment will result in this parent approval being revoked and require that the NOM revisit this approval.

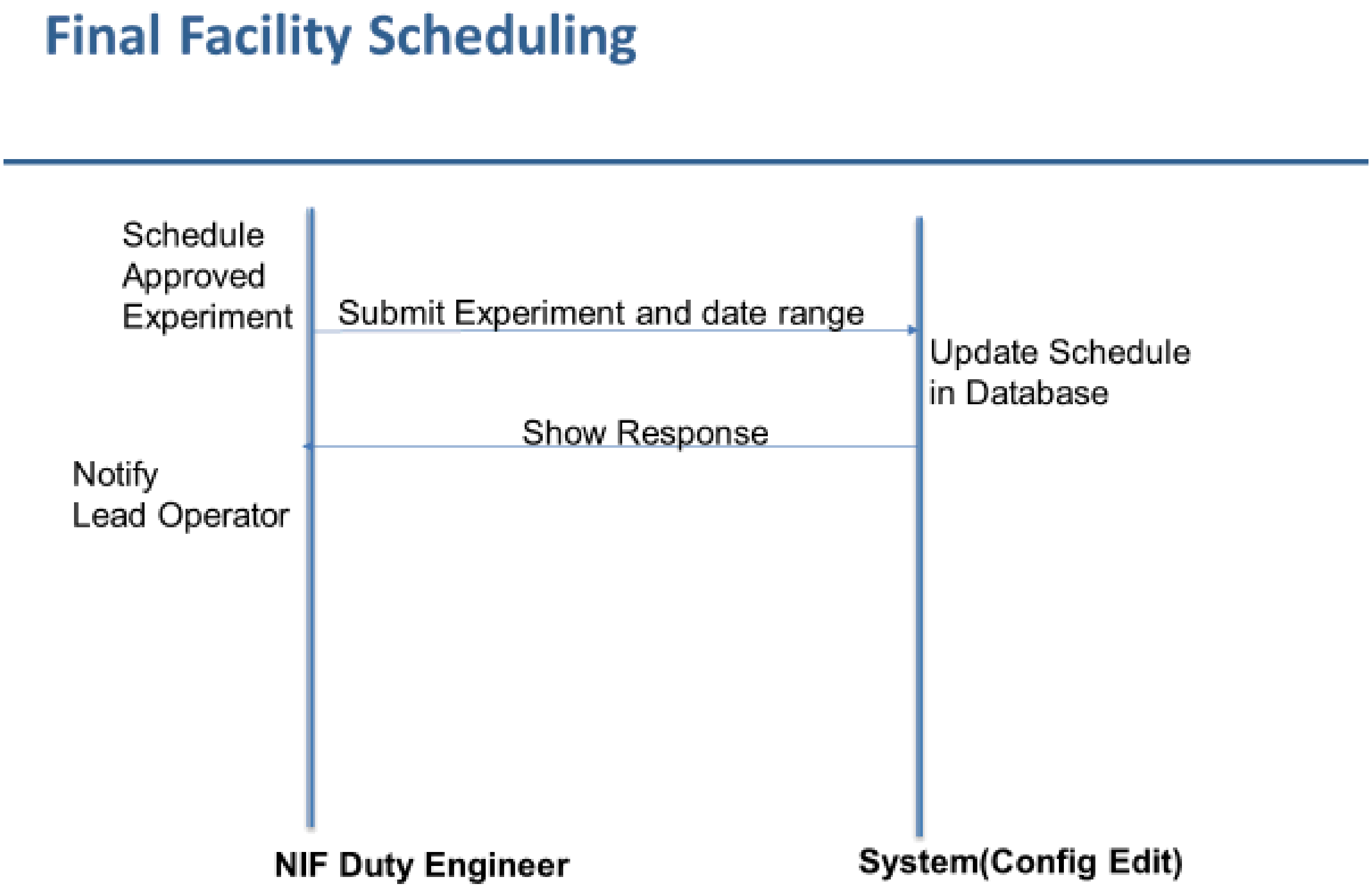
6.2.2.10 NOM Approval of Gatling Shot Sequence Diagram



6.2.2.11 Final Facility Scheduling

Here the solution remains unchanged. Scheduling a Gatling Experiment is just another Experiment ID to the Duty Engineer.

6.2.2.12 Final Facility Scheduling Sequence Diagram



6.2.2.13 Initial Loading and Execution of a Gatling Shot

Here the information generated in the GSC regarding ordering is leveraged to aid in executing all four target shots. This includes retrieving said information from "IMode" tables on shot load and modifications to Shot Setup to allow "re-aggregating" of the parent experiment to include the next target information. From there, existing alignment tools as well as updated automation will aid in firing only the bundles of interest to the next target on the next target.

ICCS will be modified such that any experiment, though approved and exported, but still annotated as being a component experiment to a parent Gatling Sequence Experiment will not be available for loading in ICCS. ICCS will be modified to recognize GATLING CMT Shot Type and manage order, bundle selection/dropping and target validation via Gatling Description provided by CMT which includes parent, children, and order. ICCS will be modified to add the Gatling Experiment ID to archiving.

Shot Setup will be modified to re-aggregate the parent experiment as directed by shot per shot advancement or dropping of component experiment(s).

Detailed Shot Automation behaviors and resulting Shot IDs:

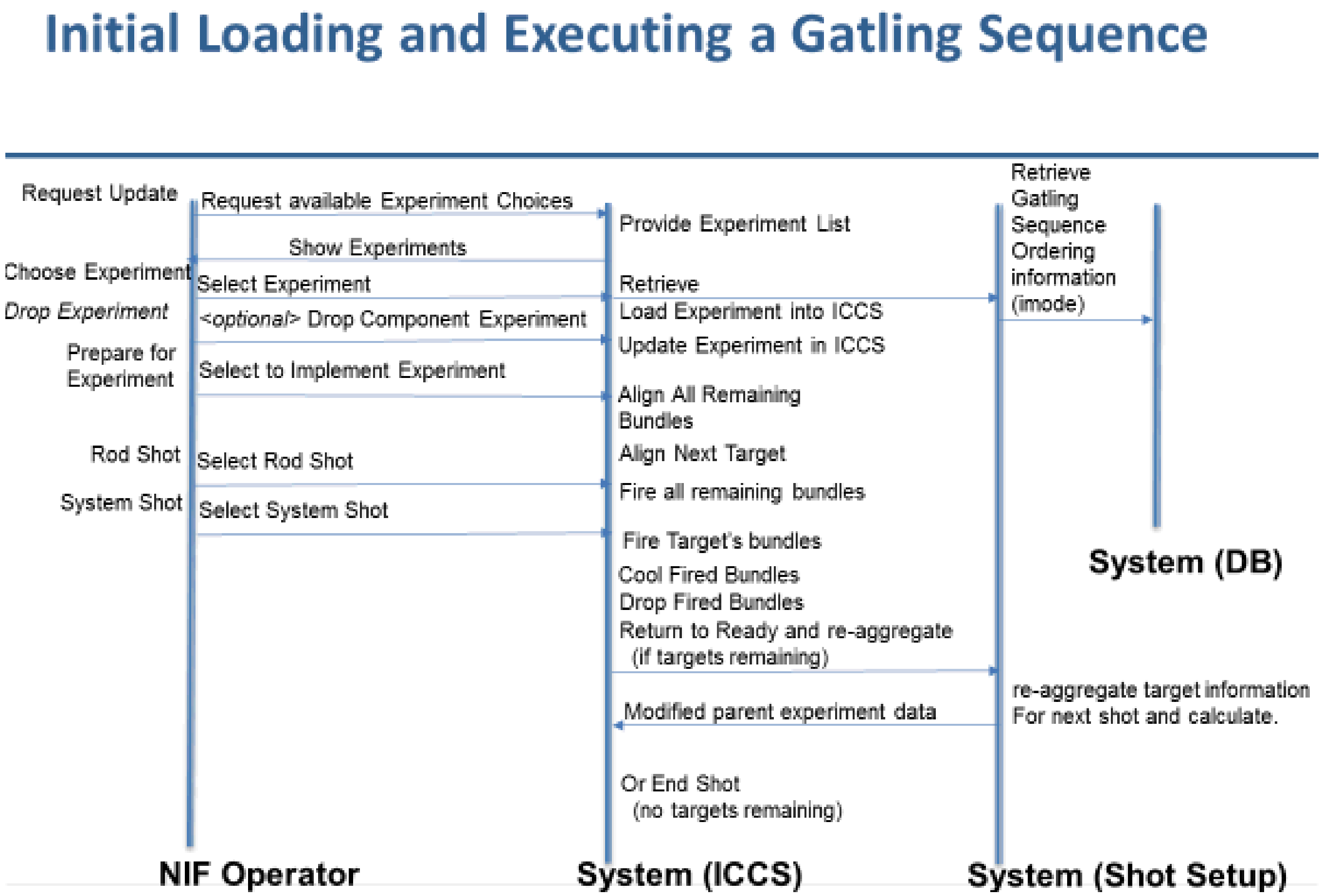
When we return to ready after a system shot in a GATLING CMT shot type
We will trigger re-aggregate for next non dropped component experiment into 001-000 in ready prior to running ready MSs,
at completion of Ready then automatically run new Target Change Flex Update System
running the new Target Change FUS will open the 002-000 archive and set 002-000 as active Shot ID
selection of new Target Change FUS first opens archive 002-000 (archive server to link active Shot ID of 002-000 to SLC ID of 001-000 in manifest)
return to ready after Auto Target Change Flex Update System (FUS)
active Shot ID returns to SLC ID,
Don't re-aggregate, don't auto FUS,
on next pre-cd do go to 002-001 (rod) or 002-999 (system)
Warning: if they select Target Change FUS when not needed it will advance to the next shot of the day.

NDATE-001-000 NDATE-001-000 Loaded Parent Gatling Experiment
NDATE-001-000 NDATE-001-000 contains dry run of first non-dropped component experiment as done during Implement Plan.
NDATE-001-000 NDATE-001-001 Rod shot on all non-dropped bundles as aggregated across all component experiments
NDATE-001-000 NDATE-001-999 System shot on first non dropped component experiment archived with Experiment ID, SLC ID, Active Shot ID, Component Exp ID
NDATE-001-000 NDATE-001-000 Returned to ready
re-aggregated to include next component experiment Target and Diag content.

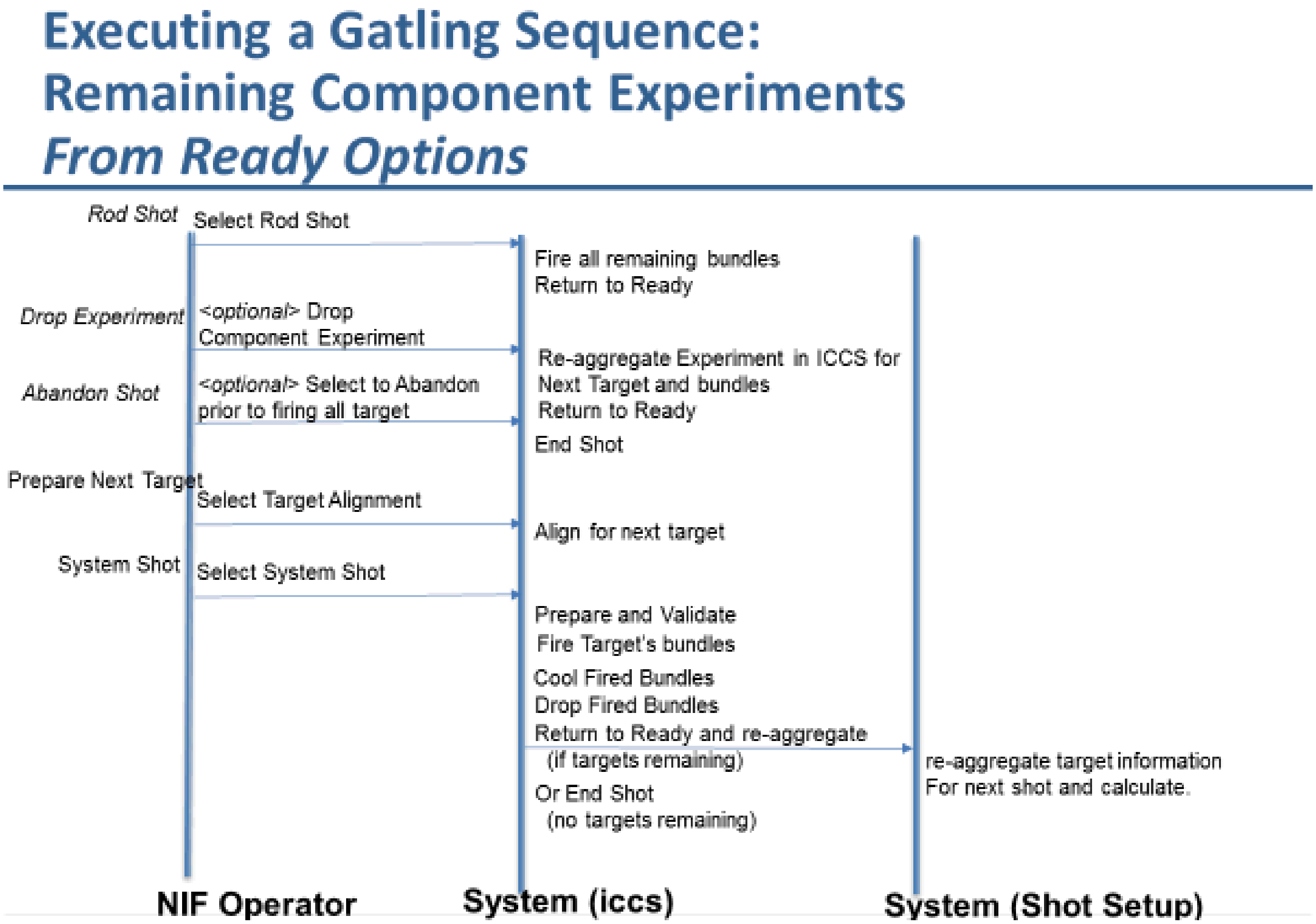
dropped shot bundles,
NDATE-001-000 NDATE-002-000 Auto run Target Change FUS
NDATE-001-000 NDATE-002-999 pre cd system
NDATE-001-000 NDATE-002-999 system shot
NDATE-001-000 NDATE-001-000 Returned to ready
re-aggregated to include next component experiment Target and Diag content.
dropped shot bundles,
NDATE-001-000 NDATE-003-000 Auto run Target Change FUS
NDATE-001-000 NDATE-003-999 pre cd system
NDATE-001-000 NDATE-003-999 system shot
NDATE-001-000 NDATE-001-000 Returned to ready
re-aggregated to include next component experiment Target and Diag content.
dropped shot bundles,
NDATE-001-000 NDATE-004-000 Auto run Target Change FUS
NDATE-001-000 NDATE-004-999 pre cd system
NDATE-001-000 NDATE-004-999 system shot
NDATE-001-000 NDATE-001-000 Returned to ready
Modify to auto End Shot.
Do not re-aggregate
NDATE-001-000 NDATE-001-000 End Shot Cycle (either by selection or auto)

Provides:
smart automated execution of order Gatling Experiments
Inputs:
Direct user input, Gatling Experiment cap data (Components and order
Outputs:
well archived component experiments
Resource Needed:
Cap Information, Parent and Component Experiments

6.2.2.14 Initial Loading and Execution of a Gatling Shot Sequence Diagram



6.2.2.15 Ready Options Gatling Shot Sequence Diagram

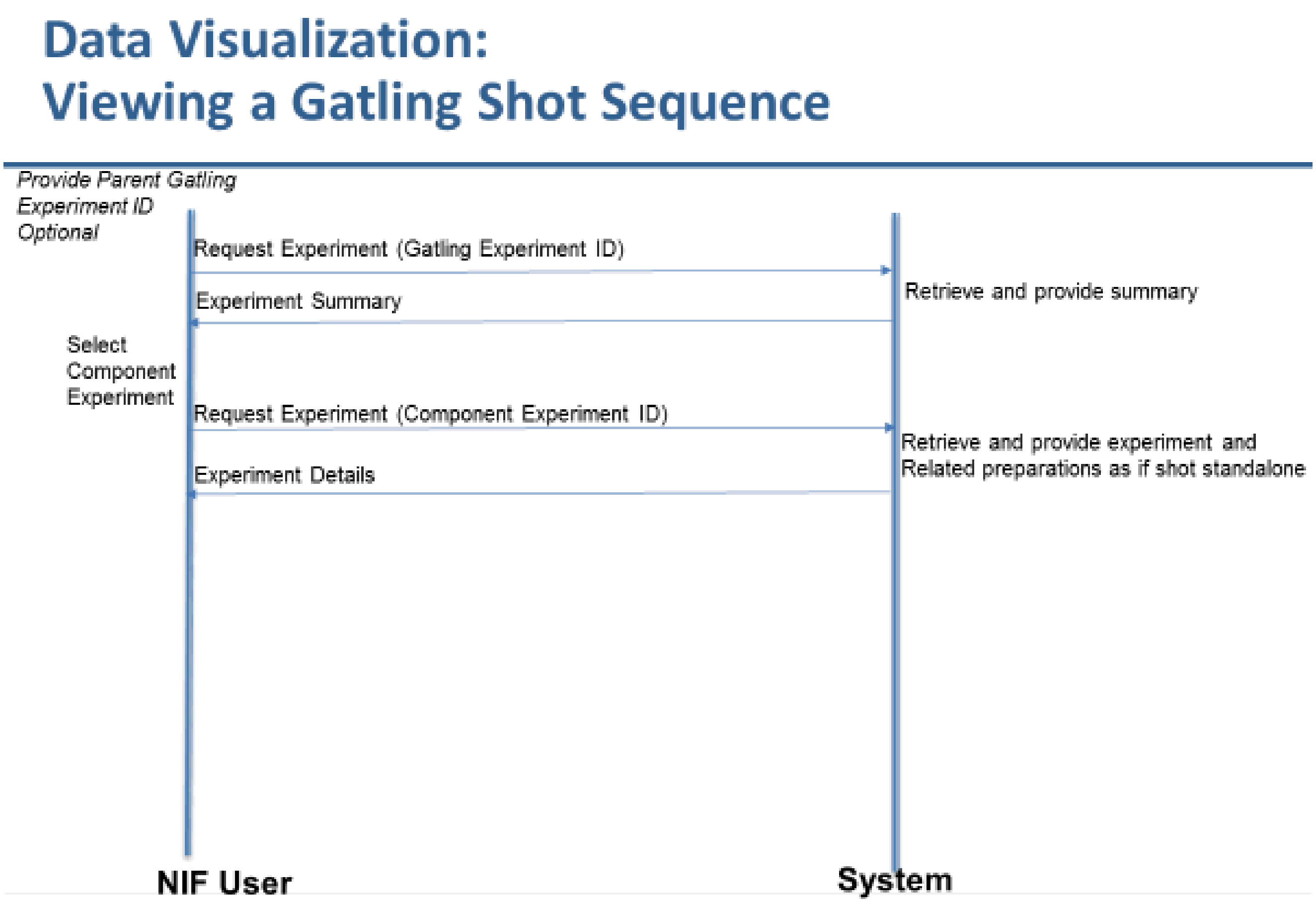


6.2.2.16 Data Visualizaion post shot review

Data Visualization will be modified to allow individual component experiment review using Shot Lifecycle ID, Shot ID, Component Experiment ID and Gatling Sequence Experiment ID as well as System Shot and Rod Shot times to ***seamlessly*** relate data. Archive viewer to allow component to parent or parent to components navigation.

Provides:
clear review of component experiment results as if done standalone
Inputs:
updated archiving to include Shot Lifecycle ID, Shot ID, Component Experiment ID, Parent Gatling Experiment ID, Rod and System Shot times and bundles fired
Outputs:
clear user interface
Resource Needed:
modified Archive Data

6.2.2.17 Data Visualization post shot review Sequence Diagram

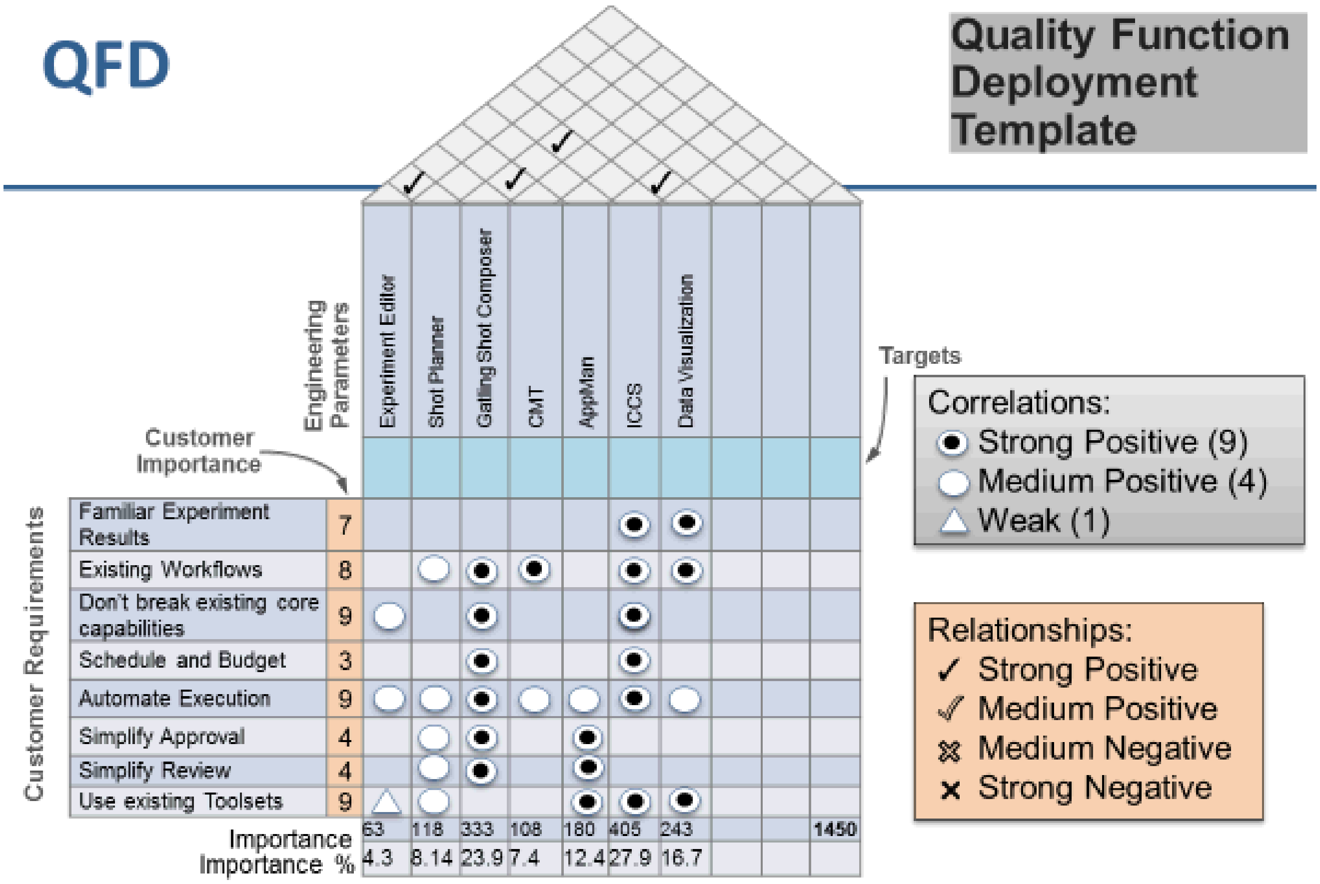


6.2.2.18 Post shot update of LPOM Target Mass information

LPOM will be modified such that in post system shot review the mass of the target is retrieved from the archive data.

Provides:
Updated method of post system shot review regarding retrieving target mass from shot archive data.
Inputs:
looking up target mass by component experiment
Outputs:
updated model
Resource Needed:
Modified Archive Data

6.3 QFD



On analysis, ICCS shows itself to be of great importance to our goals. On reflection, one could also see that activities related to Shot Planning, while important to those doing the planning and reviewing, may be treated as secondary as compared this those activities directly responsible for the shots automation.

Section 7

Proposed System Operational Architecture

7.1 Changes to the legacy architecture

At the top most level the System Architecture remains intact and unchanged. That is, Plan, Execute, Visualize.

At the next level down, architectural changes are being made in Shot Planning. This is represented by the addition of Experiment Editor to manage bulk Gatling Flip ID generation with minor interface changes to support direct Gatling Component entry. Changes which allow easy recognition of the Gatling Shots and Components are minimal as the Flip IDs themselves have been modified to include this indication.

Changes within Shot Execution are limited to interfaces and behaviors. However, the changes to Shot Execution for example require significant care and attention due to the complexity of the change and the fact that those changes are not easily separable from existing behavior. This places risk on this area that spills over to non-gatling operations. This risk is mitigated by thorough test plans and offline testing.

Changes to Data Visualization are limited to interface and behavior in its use of additional archive information to relate present shots as if stand alone.

Section 8

System Requirements

8.1 Gatling Sequence Top Level Requirements

- *The system shall allow a NIF User access to more exploratory target shots via a demonstrable decrease in the time required to execute 4 individual target shots under separate shot lifecycles. Functional*
- *The system shall allow for executing up to four target shots each at system energies for the bundles involved. Functional*
- *The system shall not allow re-firing of any bundle having been used on a system shot. Functional*
- *The system shall leverage and maintain the bulk of workflow and procedures used in setting up a standard shot. Functional*
- *The system shall be capable of generating useable target physics diagnostic data on every system shot. Functional*
- *The system shall provide clear indication of "as fired" details of every shot. Functional*
- *The system shall allow a NIF User to review experiment outcomes as if having been fired standalone. Functional*
- *The system shall provide early recognition of Gatling involved shots to aid in facility scheduling. Functional*
- *The system shall provide clear processes and tools related to Expert Group Approval, leveraging existing toolsets and training wherever possible. Functional*
- *The system shall manage and distribute parent to component and component to parent approval as to improve and clarify Gatling Shot approvals. Functional*
- *The system shall limit impact to existing staffing levels. Functional*
- *The system shall allow use of all current experiment based toolsets such as FDBL Reports and allow such tools to be used on either the parent or component experiment. Functional*
- *The system shall continue to provide awareness of upcoming shots to TD Factory, Target Fab, MOR SM etc. as regards planning for upcoming experiments. Functional*
- *The system shall provide automated shot execution of a Gatling Sequence using up to four targets. Functional*
- *The system shall allow a NIF User to drop any target and its related bundles from a gating shot sequence mid automation. Functional*
- *The system shall improve, that is, decrease the need for manual NIF Operator interventions and procedures as evidenced by simplification of the shot checklist. Functional*
- *The system shall reduce risk of error by managing and validating target, bundle and diagnostic relationships and halting misconfigurations. Functional*
- *The system shall provide and update target mass involved in component experiments. Functional*
- *The system shall not have an adverse effect on, or complicate standard NIF Experiments their scheduling, setup, approval, execution or data visualization. Functional*
- *The Experiment Editor Shall be created such that while adding "bulk" experiments up to four experiments intended to be executed via a "Gatling" technique may be related to one another as well as the "fifth" or Gatling Parent Experiment. (Experiment Editor) Non-functional*

- *Both the Shot Planner and Experiment Editor shall be modified to support entering and relating Gatling Experiments and at a glance review allow representation of an upcoming Gatling Shot via a distinguishable Gatling Sequence Experiment ID and up to four distinguishable Flip IDs where said Flip IDs indicate Parent or Component. (Shot Planner and Experiment Editor) non-functional*
    - *Per Discussions: Burr/Bond*
        - *Must be represented in text.*
            - *Component experiment to include "gat" in Flip ID.*
            - *Parent Gatling Experiment include "gatgr."*
- *The Shot Planner shall allow a user to hover on any component to display a tool tip indicating sibling components (Shot Planner). non-functional*
- *The Shot Planner shall allow a user to choose existing parent or create a new one when relating a component experiment to a parent.*
- *Selection of a Gatling Sequence's component experiments* ***shall*** *be limited to experiments from a single campaign. (Shot Planner)*
    - *Per Discussions: Burr/Bond*
- *A Gatling Sequence Experiment shall allow for 2 – 4 System Shots in one Shot Lifecycle. Functional*
    - *(inferred per bundle use validation GSC).*
- *Component experiments of a defined Gatling Sequence shall retain their ability to be reviewed, approved, and configured for individually. non-functional*
- *The Shot Planner shall be modified such that a New association, Re-association or Dis-association of any component experiment causes invalidation of the parent Gatling Sequence Experiment. (Shot Planner) Non-functional*
- *Shot Director s/w shall not allow loading of an individual component experiment. (Shot Director) Functional*
- *A standalone Gatling Shot Composer will be created which shall validate component experiment compatibility, define the order of their execution and aggregate/build a parent Gatling experiment from the components. (Gatling Composer) Non-functional*
- *Validation of a Gatling Experiment shall include: (Gatling Composer) Functional*
    - Bundle:
        - *Use of any Quad or Bundle within a Cluster constitutes use of that Cluster and a Cluster may not be used in more than one of a Gatling Experiments Component experiments.*
    - Diagnostic:
        - *Diagnostic usage is validated at the individual experiment level and no specific aggregation or validation is needed.*
        - *Setup for any used diagnostic must be the same in all component experiments in which it is used.*
        - *Diagnostics used in component experiments may differ experiment to experiment.*
            - *This indicates that dry runs are informed by "next up" component experiment content.*
            - *This needs to update aggregation details elsewhere.*
        - *However, we should consider two stretch goals during the approval of a Gatling experiment.*
            - *Minimize the number of reconfigurations.*
            - *Warn of diagnostic use conflicts related to classification.*
    - Target: TAS Position
        - Component Experiments have the same TAS Positions.
            - All beam TPs are less than the Target TP.
            - The Target TP is last and the same for all.
- *Once determined as valid, the Gatling Composer shall be capable of aggregating Gatling component experiments into a parent Gatling Experiment suitable for laser setup, LPOM Calculation. (Gatling Composer) Non-functional*
- *Once determined as valid, the Gatling Composer shall allow a NIF User to define the order of its Components. (Gatling Composer) Non-functional*
- *The system shall allow loading and executing of parent Gatling experiments. Functional.*

- *The system shall provide all pre and post shot behaviors including but not limited to FDBL planning are supported as would any standard shot on the new aggregated Gatling Experiment. (System) Functional*
- *Gatling Sequence Experiment Approval shall be the same as that of any standard or component experiment with the exception of a detection of any component experiment not yet having already approved said item. (AppMan)*
- *Where a component experiment's expert group review has not been approved selection of the expert group approval at the parent Gatling Sequence Experiment level shall provide a Warning/Override capability indicating that approval will be applied to the unapproved components. (AppMan)*
- *Every component experiment shall be annotated with its parent Gatling Sequence Experiment ID. (GSC, CMT/SRE)*
- *Any experiment, though approved and exported but still annotated as being a component experiment to a parent Gatling Sequence Experiment shall not be available for loading in ICCS. (ICCS)*
- *New association, Re-association or Dis-association of any component experiment shall cause a withdrawal of the parent Gatling Sequence Experiment. (AppMan)*
- *Withdrawal an expert group approval on any component experiment shall invoke withdrawal of that approval on the parent Gatling Sequence Experiment. (AppMan)*
- *Approval of a Gatling Sequence Experiment shall require validation of its component experiment's bundle usage and target compatibility. (GSC, AppMan)*
- *A Gatling Sequence Experiment shall be approved, configured for, and loaded as would any individual experiment by selecting its Gatling Sequence Experiment ID. (ICCS)*
- *Once CMT has associated a real experiment to a parent Gatling Sequence that experiment shall be available in AppMan for review including FDBL usage. (where)*
- *Gatling Shot Composer shall create a 5th experiment per the associations provided under the "gatgr" parent Gatling Experiment Description aggregating its Laser information and providing that LPOM Calculations are performed in support of expert group review and assigning it a CMT Shot Type of GATLING. (Speck/Shaw) (GSC)*
- *ICCS details*
    - *A Gatling CMT Shot Type experiment once loaded shall: Non-functional*
        - *Preload active Target information from its lowest order component experiment into the Shot Lifecycle ID context.*
    - *When executing ICCS shall provide a Gatling "return to ready…" behavior for CMT Shot Type: GATLING experiments.*
    - *When executing ICCS shall return CMT Shot Type: SYSTEM experiments to auto end shot after system shots.*
    - *When executing a Gatling Sequence experiment ICCS shall automatically maintain order, target information and bundle usage of its executing component experiments.*
        - *In execution of a Gatling Sequence, completion of an executed component experiment shall automatically re-aggregate the next Target Information into the Shot Lifecycle ID context.*
    - *In execution of a Gatling Sequence, an executed component experiment shall*
        - *Archive its Shot IDs relations to:*
            - *Shot Lifecycle ID to (existing)*
            - *System Shot ID to (existing)*
            - *Component Experiment ID to (existing)*
            - *Gatling Sequence Experiment ID . (new)*
    - *In execution of a Gatling Sequence Experiment, an executed component experiment shall be dropped at return to the Ready State thus automatically dropping the bundles just shot.*
    - *A Gatling Sequence shall support dropping any component experiment.*
    - *On dropping of a component experiment the bundles defined for that experiment shall be dropped.*
    - *In execution of a Gatling Sequence, a dropped component experiment shall be skipped in the ordered execution of the remaining Gatling Sequence System shots.*
- *Data Visualization shall allow NIF User to view and component as if it had by executed standalone including all rod shots relevant to its bundles. Non-functional*
- *Data Visualization shall allow a NIF User to access shot information via the parent's Experiment ID, Shot Lifecycle ID, component Experiment ID, or component Shot ID. Non-functional*

- *A Gatling Sequence shall support FDBL usage by any or all of its component experiments. Non-functional*
- *FDBL Usage in a Gatling Sequence shall be managed under the Gatling Sequence ID. Non-functional*
- *FDBL Activities previously accessed and supported by Experiment ID shall also be accessed and supported by Gatling Sequence ID as if a single experiment. Non-functional*

Section 9

Functional Architecture

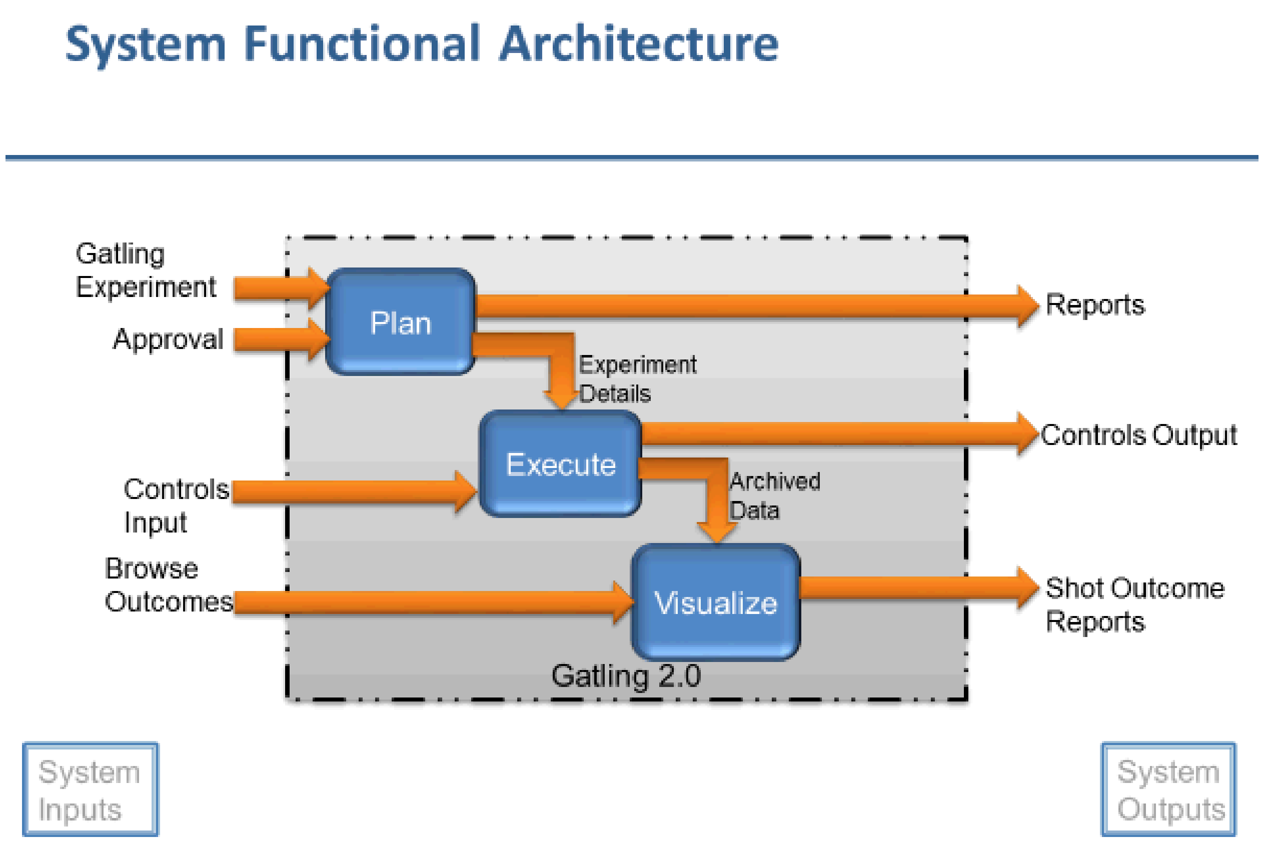
9.1 Functional Architecture

While the functional architecture appears unchanged at the top level, the modification required to achieve the Mission Statement are distributed within these sub functions and those systems requiring new interfaces and functionality have been detailed above.

Plan has been updated to provide simple bulk input to users without impacting traditional users and provide validation and ordering of component experiments.

Execute has been updated to leverage ordering to automate firing, archive as fired.

Visualize has been updated to browse Gatling sequence and view any Gatling fired shot as if fired standalone.



Section 10

Organizational and Business Impact

10.1 Business impact

10.1.1 Legacy processes

This system strives to minimize impacts on legacy processes while also maintaining a cohesiveness of function among existing systems. Items such as adding of an Experiment Editor allows power users of Gatling to input shot place holders easily without impacting methods and tools already familiar to core NIF users. This theme is also represented in the addition of a Gatling Shot Composer which provides validation, aggregation, and ordering by producing an aggregated shot setup. This approach means that firing a Gatling Shot looks very much like any other shot, and those areas of greatest risk are automated around using existing safeguards such as not re-firing a bundle.

10.1.2 Mission Impact

A NIF User understands clearly the value of facility time. Wherever possible it is in all our interests to make that time as efficient and effective as possible. During this process care was taken to interview and assess any additional burdens placed on each phase of the process. As such we have achieved stakeholder “buy-in” regarding the proposed approach and confidence in our managing and minimizing risk or impact on existing shot reliability.

Once this system is delivered and functioning, it is estimated that four target shots would be executed in less than half the time they would had they been planned and run individually. Its comparison with Gatling 1.0 must account for having automated and managed the risks and concerns described by users regarding the prototypical complexities. The true return on investment is subject to our socialization of the capability and its acceptance by the user community. While the modified tools and approach would be beneficial to any single Gatling Shot in easier setup, execution, and understanding of outcomes, broader mission impact hinges on a user community which embraces the experimental approach and plans the exploratory shots to leverage it.

Section 11

Risks and Technology Readiness Assessment

11.1.1 Technical Risk Matrix

Key areas of risk are related initial roll out. Maintaining all existing functionality.

11.1.2 Risk Matrix

	Negligible	**Marginal**	**Critical**	**Catastrophic**
Certain		Shared delivery schedules		
Likely	Training Issues			
Possible		Creating Invalid Experiments	Creating Unsafe Experiments	
Unlikely			Break Legacy System(s)	Re-firing shot bundles
Rare			Firing wrong Bundle to Target	

11.1.3 Failure Modes and Mitigation

Failure	**Mitigation**
Break Legacy System(s)	Where possible minimize code and procedural intrusion Offline and Online regression testing
Firing wrong bundle to wrong target	Use validated aggregate information from reviewed component experiments coupled with automation to avoid user errors Offline and Online regression testing
Creating invalid	Multi-layer review and approval Offline and Online regression testing
Creating unsafe	Multi-layer review and approval Offline and Online regression testing
Training Issues	Where possible use existing procedures and/or logical and expected approaches Provide offline and online training Socialize widely

Section 12

Reflective essay

12.1 Lessons learned

While the Mission Statement reflected exactly the needs of the customers, the breadth of a system such as NIF requires very active discussion with a range of internal customers. Formalizing Gatling Shots in a system as mature and yet complex as NIF in and of itself is a challenge. I found myself striving for the customer's need, but also striving to defend those customers who never intend to execute this kind of shot. It quickly became clear that every phase of this system and subsystem needed to consider the "non-Gatling customer" as well. The bulk of our users who will not use this capability must be represented in its development so as to avoid adversely affecting their experience.

In addition, these approaches were always discussed in the "real world", where schedule, budget, and skillsets available were weighed to determine: where a function should live: if placed there, could it be developed and supported in a shared timeframe; and whether it would be maintainable.. An engineering approach which would stand no chance of being built would not be considered a solution; it would just be an idea.