



Project Acronym	Fed4FIRE
Project Title	Federation for FIRE
Instrument	Large scale integrating project (IP)
Call identifier	FP7-ICT-2011-8
Project number	318389
Project website	www.fed4fire.eu

D8.4 - Second input to WP2 concerning first level support

Work package	WP8
Task	T8.1
Due date	30/09/2013
Submission date	27/09/2013
Deliverable lead	Dai Davies (DANTE)
Version	1.0
Authors	Dai Davies (DANTE)
Reviewers	Wim Vandenberghe (iMinds), Piet Demeester (iMinds)

Abstract	This document provides an update on the implementation of First Level Support (FLS) and defines additional requirements that FLS will have on the project.
Keywords	First Level Support, FLS, Operations, Trouble Ticket,

Nature of the deliverable	R	Report	X
	P	Prototype	
	D	Demonstrator	
	O	Other	
Dissemination level	PU	Public	X
	PP	Restricted to other programme participants (including the Commission)	
	RE	Restricted to a group specified by the consortium (including the Commission)	
	CO	Confidential, only for members of the consortium (including the Commission)	

Disclaimer

The information, documentation and figures available in this deliverable, is written by the Fed4FIRE (Federation for FIRE) – project consortium under EC co-financing contract FP7-ICT-318389 and does not necessarily reflect the views of the European Commission. The European Commission is not liable for any use that may be made of the information contained herein.

Executive Summary

First Level Support (FLS) is one of the central functions of the Fed4FIRE federation. It provides a common facility for the logging and resolution of faults. It provides a direct line of communications with experimenters and with the support functions in the individual test-beds. This document considers how the initial architectural requirements of FLS have been met and describes the processes that FLS intends to use in the operational phase of Fed4FIRE. It considers how, in a live operational environment within the federation, operational support is likely to function. Given the absence of any operational data today, it uses, as a proxy, experience gained from a similar federated operational environment in the sector of backbone networks. Using this analysis a number of proposals are made for the way in which the operational architecture of the federation can be developed.

Acronyms and Abbreviations

FLS	First Level Support
KEDB	Known Errors Database
SME	Subject Matter Expert
TTS	Trouble Ticket System



Table of Contents

- 1 Introduction 7
- 2 Current State of FLS Implementation 9
- 3 New Architectural Requirements – Background 10
- 4 New Architectural Requirements – Implications 12
 - 4.1 Test-Bed Operational Practices..... 12
 - 4.2 Operations in a Federated Environment 12
- 5 Current State of Test-Bed Operational Functions..... 13
- 6 New Architectural Requirements – Obligations 14
 - 6.1 Organise Pro-active Announcements of Maintenance..... 14
 - 6.2 Establish Appropriate Subject Matter Expert links 14
 - 6.3 Create Agreed Operational Best Practice Code 14
 - 6.4 Harmonise Maintenance Windows 14
- Appendix A: Fed4FIRE First Level Support Processes..... 16



1 Introduction

Deliverable D8 .1 introduced the concept of First Level Support (FLS). FLS is responsible for the logging and resolution of operational issues associated with the Fed4FIRE federation of test-beds. Figure 1 below illustrates the key components of FLS.

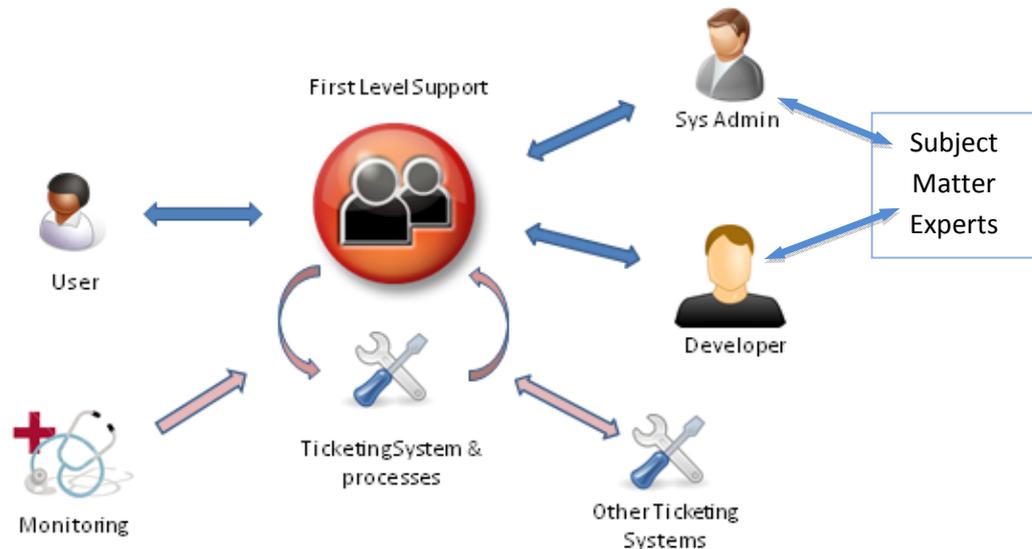


Figure 1 Schematic of First Level Support

The main operational system that FLS uses is the Trouble Ticket System (TTS). The TTS is a central database in which all operational problems, associated with the Fed4FIRE federation, are logged. The other major element of FLS is the set of processes used to control the resolution of problems. FLS is a very process driven approach. Standard processes are defined for the functions carried out by FLS. These are described further in Appendix A below. The processes are initiated either

- i. as a result of problems detected by the Fed4FIRE monitoring system which provides FLS with an overview of the functioning of the individual test-beds within the federation or
- ii. as a result of interactions between FLS and the equivalent support functions within the individual test-beds of the federation or
- iii. following a request from an experimenter

An important concept, within FLS, is that of the Subject Matter Expert (SME). SME's have particular, in-depth knowledge of the architectural components that constitute the federation and/or detailed knowledge of the functioning of an individual test-bed.

FLS is staffed by engineers whose principal skill is problem analysis/resolution. They do not have detailed knowledge of the technical elements of the test-beds within the Fed4FIRE federation. Nor do they have detailed knowledge of the particular operational processes within an individual test-bed. As a result, where they encounter a technical problem associated with a particular test-bed, which they have not previously experienced, the FLS staff rely on the appropriate SME to deal with the detailed problem analysis. In this case FLS supervises and documents the fault resolution process.

Another important concept, which is part of FLS, is The Known Errors Database (KEDB). The Known Errors Database contains details of problems, which are already known to exist, and where a work-around has already been identified. This database enables FLS staff to resolve problems without having to directly involve SMEs.

In D8 .1 the basic architectural requirements of FLS were identified. This deliverable describes in more detail the planned implementation of FLS. It considers the way that their implementation has met the objectives of those requirements. It also identifies new architectural requirements which will assist the further development of FLS.

2 Current State of FLS Implementation

D8 .1 recognised the key architectural requirement of a ‘monitoring capability’ (dashboard) that would indicate in a “near real-time basis” the status of the key components of individual test-beds within the federation. The dashboard would enable FLS operators to detect operational problems and initiate actions to resolve problems in a proactive way. This dashboard has already been implemented and provides the following indicators.

- i. Network Reachability of a test-bed as shown by “ping” measurement
- ii. Indication of the functioning of the test-bed’s Aggregate Manager by testing the GetVersion status, thereby giving a simple indicator of availability.
- iii. Summary of free resources on an individual test bed. This enables FLS to easily detect requests that exceed this resource limit.
- iv. Integration of the internal monitoring of a test-bed to provide a summary health status.
- v. Indication of when the indicators were refreshed. The test-beds are polled to obtain the data

A screenshot of the dashboard is shown in figure 2.

Fed4FIRE First Level Support Monitoring					
Testbed Name	Ping latency (ms)	GetVersion Status	Free Resources	Internal testbed monitoring status	Last check internal status
BonFIRE	19.28	N/A	N/A	ok	2013-07-05 09:00:49+02
Fuseco	26.76	ok	2	ok	2013-07-04 22:46:04+02
NETMODE	67.15	ok	20	ok	2013-07-05 09:00:22+02
NITOS	72.75	ok	31	N/A	N/A
Planetlab Europe	31.46	ok	298	N/A	N/A
SmartSantander	47.34	ok	0	ok	2013-07-05 09:00:01+02
Virtual Wall	0.16	ok	31	ok	2013-07-05 08:56:39+02
w-iLab.t2	2.4	ok	53	ok	2013-07-05 08:56:58+02

Figure 2 Screen shot of the FLS monitoring tool (Dashboard)

The dashboard was demonstrated at the project review meeting in Ghent in July 2013. The FLS service is only due to go live in month 17 of the project (February 2014). It is likely that enhancements to the dashboard will be requested as a result of operational experience. Currently, the dashboard meets the basic needs of FLS.

The other key components of FLS are the set of processes that FLS adopts for managing issues and the Trouble Ticket System. These are described in more detail in Appendix A.

3 New Architectural Requirements – Background

The current Fed4FIRE architecture deals with the overall building blocks necessary to create the federation of test-beds. It does not, in general, go into the detail of information flows between the building blocks. To this extent it is a static picture.

The architecture captures the main requirements of FLS in its interactions with the test-beds. It does not really address the information flows between FLS and equivalent test-bed specific support functions. These will be, for the foreseeable future, defined by a set of processes and e-mail exchanges as the basic method of implementation. It would be possible to consider much greater operational integration among the test-bed support functions, including a federated operational structure, but these are research topics in their own right and, both from a resource and also from a scope point of view, are somewhat outside the current description of work of Fed4FIRE.

Nevertheless, an important architectural area not yet explored is the interaction between FLS and the support functions of the individual test-beds in the Federation. We do not yet have operational experience of supporting a federation of test-beds but there is a significant analogous experience to be derived from the operational aspects of GÉANT (the pan-European research and education network). GÉANT interacts with connected national research networks in a way which is similar to the proposed Fed4FIRE federation. In order to develop federated operations in fed4Fire the experience gained over many years in GÉANT has been used as a proxy.

Analysing the type of problems detected in GÉANT as recorded in the Trouble Tickets gives an insight into the likely pattern of problems to be expected in Fed4FIRE. Table 1 and an associated Pi chart (figure 3), below, show such an analysis of trouble tickets recorded in the GÉANT network operations Centre (analogous to FLS). The table illustrates the number of tickets recorded over a recent six-month period categorized by the type of event being dealt with. The event types are defined as follows.

- **Incident** - An unexpected problem either detected by monitoring or reported.
- **Maintenance** – An expected interruption to service which may be scheduled significantly in advance or maybe as a result of emergency action
- **Service Requests** – Requests for additions to or changes to services provided
- **Housekeeping** – Use of the Trouble Ticket system for internal issues associated with the network operations function itself
- **Security** - Tickets associated with security problems
- **Escalation** – A serious complaint about an aspect of service. Escalation can also be used proactively to note major incidents detected.

Type of Ticket	Ticket Count
Incident	709
Maintenance	894
Service Requests	94
Housekeeping	42
Security	13
Escalation/Complaint	1

Table 1 Analysis of GÉANT trouble tickets by type

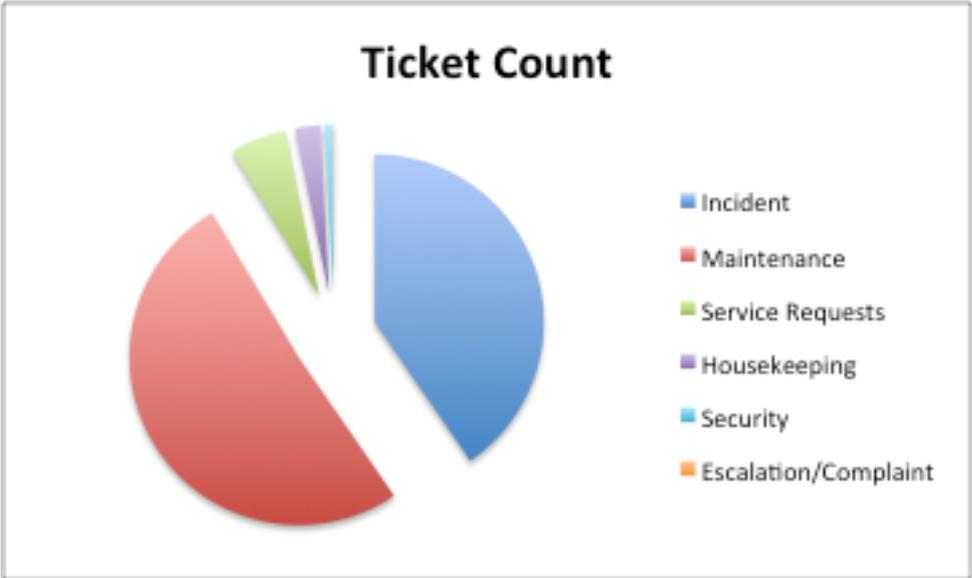


Figure 3 Ticket Analysis Pi chart.

The table shows that, although incidents are a significant part of the total problem set being resolved, the most significant category recorded is maintenance tickets. These tickets are related to standard operational maintenance. In the case of GÉANT these relate to maintenance activities of the connected networks or of the GÉANT network itself. Most of this maintenance, approximately 75% is scheduled maintenance. It is reasonable to assume that a similar pattern will be found in the federation of test-beds.

4 New Architectural Requirements – Implications

4.1 Test-Bed Operational Practices

Operational support provided by individual test-beds in Fed4FIRE today takes place in a closed environment. Experimenters have a direct relationship with an individual test-bed. Lines of communication between a test-bed operator and an experimenter are quite short. An informal approach to communication of scheduled outages is probably adequate. This approach to support will not scale to a federated environment where there is a much weaker link between an individual test-bed and an experimenter.

4.2 Operations in a Federated Environment

In the context of the Fed4FIRE federation, the existence of FLS, and the presence of a monitoring dashboard will mean that an outage schedule by a test-bed will appear as an alarm on the FLS dashboard. This would, ordinarily, lead to the creation of an incident trouble ticket and initiation of a fault resolution process. Because scheduled maintenance is such a regular event, the operational processes implemented by FLS are designed to deal with it. This is initially achieved by opening a trouble ticket associated with scheduled maintenance in advance of the maintenance taking place. When an alarm condition is detected on the dashboard the FLS operator first checks to determine whether this alarm is associated with a maintenance and then, only if no maintenance has been scheduled, will the full problem resolution process be invoked.

The requirement to operate in a federated environment necessarily leads to a new obligation on a test-bed for the active provision of operational information to FLS. It is this aspect of federated operations that defines the additional architectural requirements. The architectural requirements identified here relate to the information flows and interactions between FLS and the individual test-bed operational functions. As such, they have no practical implications for the technical architecture that has been previously developed in WP2. They nevertheless give rise to operational architecture requirements which define the operational interactions that are required between individual test-bed operations and FLS.

The focus of Fed4FIRE is to create a federation of interconnected test-beds. This federation will have implications for the operational functions of individual test-beds. It is not an objective of the project to develop a federated operational environment per se. The architectural implications of the operational federation are, therefore, limited to defining operational interactions between test-beds and FLS and developing statements of operational best practice. The best practice will be implemented by the test-beds using their existing operational support systems and developing common and harmonized approaches to operations.

5 Current State of Test-Bed Operational Functions

As noted in Section 4 existing test-beds tend to operate in a fairly closed environment and, typically have a direct relationship with their experimenters. Operational support in such an environment can be organised informally.

Operational support can be characterised by a number of parameters. The most significant ones are:-

- i. **Availability of support.** The period of time during which such support is available and active. For FLS in Fed4FIRE it is planned to offer support during core working hours [0900-1700 CET] Monday – Friday.
- ii. **Time to respond to/resolve a problem.** The time taken to resolve a problem is obviously dependent upon the nature of the problem. There is a requirement to have an initial triage function that will analyse the nature and severity of the problem. Precise categories will be test-bed specific but it is highly desirable that a common vocabulary is developed in Fed4FIRE as well as common time targets for problem identification and resolution.
- iii. **Monitoring/Performance Reporting.** Operational support functions generally have the opportunity to gather performance related information. Much of this can be derived from information stored in trouble ticket systems. By synthesizing this in summary reports it is possible to determine trends in problems and provide a tool to assist in forward planning

In order to understand the current status of operational support within the test beds WP 8 Task1 carried out a review of existing support arrangements. The review showed:

1. Many test-beds did not have stated service levels.
2. There were no stated or target response times
3. Best/Reasonable Efforts were offered for most support functions
4. There was no performance monitoring of the support function itself
5. There was no performance reporting of individual test-beds.

In order to develop federated operations in Fed4FIRE it is necessary to take into account this historic context and to recognise that the primary objective of the project, namely to facilitate the technical federation of the participating test beds. Nevertheless, there are a number of practical steps that can be taken to develop a common operational framework integrated with FLS. These are described in Section 6.

6 New Architectural Requirements – Obligations

The key elements, which are required to create a common operational environment encompassing the test-beds and FLS, are to standardise information flows and to define and develop a climate of operational best-practice. Specific proposals are described below.

6.1 Organise Pro-active Announcements of Maintenance

It is expected that maintenance will represent a significant element of the operational issues affecting the federation. As demonstrated in Section 4.2 it is vital that such maintenance, which is usually scheduled, should be registered in advance in the FLS TTS. Systematically doing so, will avoid outages caused by maintenance being diagnosed as incidents. Automated mechanisms that allow test-beds to easily communicate scheduled maintenance to the FLS as part of their operations is an important architectural requirement. Even where maintenance is unscheduled, logging the maintenance in the TTS directly prior to starting work, will be effective. All test-bed operational functions shall have an account on the TTS and can log in and raise the necessary ticket.

6.2 Establish Appropriate Subject Matter Expert links

It can be seen from table 1 that, for GÉANT, approximately half of the trouble ticket opened are associated with pre-planned operational work, the other half being associated with spontaneously occurring technical problems. In the former case, FLS will need to interact with operational staff within the test-bed, whereas, in the latter case, the interaction is equally likely to be with developers, either in the project, or in a specific test bed. Both test-bed operational staff and developers are SMEs. But there is a distinct difference between their skill sets even though, in some cases, they may be one and the same person. In order to effectively develop appropriate lines of communication FLS would need to develop contacts with the two types of SME identified above. In particular, for test-beds, this will require the identification of an appropriate operational support person or function to act as an ‘operational’ SME for that test-bed. There is a need to develop mechanisms to allow test-beds to automatically provide and maintain information about the identity of their operational SME’s

6.3 Create Agreed Operational Best Practice Code

Section 5 describes the high level parameters that define the performance of operational support from the perspective of an experimenter. It is necessary to harmonise their definition among the test-beds and with the FLS function and to agree performance targets as part of a **code of best practice**. It is recognized that not all test-beds will be able to develop their operational capabilities immediately. To encourage compliance, it is proposed that the code should recognize differing levels of conformance, with a minimum level to which all test-beds would be expected to comply, an intermediate level encompassing enhanced operational capabilities and a top level where all parameters of the code would be met.

6.4 Harmonise Maintenance Windows

For an individual test-bed, it is relatively unimportant exactly when maintenance is scheduled and performed. It is good practice to have a pre-defined window so that experimenters’ expectations can be managed. In a federation of test-beds, unless the timing of maintenance windows is harmonised,

maintenance may have a damaging effect on live experiments using multiple test-beds. Lack of a common approach to maintenance will, in any case, limit the availability of the federation.

Appendix A: Fed4FIRE First Level Support Processes

FLS is based on a series of pre-defined processes. The processes proposed for Fed4FIRE are

1. Information request where FLS is responding to a general query about Fed4FIRE
2. Service Request where FLS is responding to a specific request for a service
3. Problem Management where FLS manages the logging and resolution of a detected or reported problem.

The following descriptions outline how the processes work. It is followed by Flow Charts illustrating these processes. The Problem Management process is split into its constituent elements, Problem Logging (Core Activity) and Problem Resolution.

References given in braces link to examples and illustrations (in section 8) on how to user JIRA.

In the activities described below all steps are taken by the FLS unless specifically shown otherwise.

A) Description of the Processes (see corresponding flowcharts at the end of this document)

A-1) Fed4FIRE FLS Core Activity (see flowchart on p. 25)

Trigger event: Receipt of JIRA Ticket

Entitled Fed4FIRE users will e-mail helpdesk@fed4fire.eu with their request or concern. This will automatically create a ticket in the JIRA Fed4FIRE project and a mail notification will be sent to FLS. Go to 1.1

Trigger event: Monitoring System Alarm

FLS team observe alarm on the Fed4FIRE Monitoring System and/or an e-mail notification is received from the Monitoring system. FLS team create an Incident ticket in JIRA

1.1 Categorise Ticket

For reported issues, change the ticket category from 'Helpdesk' to the appropriate category ('Incident', 'Service Request', 'Information Request')

Incident – any actual or suspected fault or failure of a Fed4FIRE system or participating test bed.

Service Request – a request for service e.g. access to a specific test-bed

Request for Information – a question not related to a fault or the provision of a new service.

Assign the ticket to yourself and go to 1.3

1.2 Create JIRA ticket

Create an 'Incident' JIRA ticket with a title ('Summary') in the form "<test-bed name> <attribute> <condition>"

Assign the ticket to yourself and go to 1.3

1.3 Existing Issue?

Check TTS to see if this issue is related to an on-going incident. If it is then go to 1.4 else go to 1.5

1.4 Link tickets and place on hold

Add a comment to the ticket and link the newly created issue with the existing issue (see "Linking Jira tickets" section of this appendix). Place ticket on hold until the original issue is fixed. Go to 1.5

1.5 Resolved by fixing linked issue?

Resume the ticket with a note that 'Linked issue xxx has been fixed'. If this issue was resolved by fixing the pre-existing issue then go to 4.1 else go to 1.6

1.6 Update ticket

Add a comment that this issue has not been successfully resolved by fixing the other issue and go to 1.3

1.7 Is this an incident?

If this is an 'Incident ticket', go to 1.8

If this is a Service Request go to 3.1

If this is a Request for Information go to 4.1

1.8 Known Error?

Check the symptoms described against the Known Errors Data Base (KEDB) to determine if this is a Known Error. If necessary request the user to provide more information. If the issue is determined to be a Known Error then go to 1.9 else go to 1.11

1.9 Implement workaround

Implement the known work around as described in the KEDB. Add a Comment to the ticket to describe the action taken.

1.10 [User] Issue resolved?

If the user confirms the issue appears resolved then go to 4.1 else go to 1.13

1.11 Dashboard Alarm?

If this ticket is related to a dashboard alarm go to 1.12 else go to 1.13

1.12 Update ticket and escalate

Add any additional relevant information to the ticket and escalate it by assigning it to the SME indicated at section "Fed4FIRE Subject Matters Experts" of this appendix. Go to 1.14

1.13 Collect Full Information and escalate

Gather full information from the person reporting the issue. This should normally include, but not be limited to:

- Which test-bed(s) are involved?
- When did the problem start?
- How did the user become aware of the problem?
- When was the service last known to be working correctly for the user?
- What behaviour did the user expect to happen vs. what actually happened?
- Has the user experienced this problem before?
- Is the user aware of any changes made prior to the issue occurring?
- What environment (OS etc.) is being used?
- Are any other users currently experiencing this problem? Are any other users currently working unaffected? [Get contact details]

If any other Fed4FIRE users need to be kept informed of progress on this ticket then add them as watchers of the ticket {see section “Add watchers to a ticket” in this appendix}. Go to 1.14

1.14 [SME] Investigate and Resolve

The SME will work with whoever is required (the person reporting the problem, other Fed4FIRE users, other SMEs) in order to resolve the issue. The SME will add comments to the ticket to describe progress and actions taken, and if appropriate will re-assign the ticket to other SMEs to take responsibility for. Once the user has confirmed the issue is resolved go to 4.3

1.15 [User] Issue resolved?

If the user confirms the issue appears resolved then go to 4.3 else go to 1.14

A-2) Response to Information Requests (see flowchart on p. 26)

2.1 Is answer known?

If the answer to the question is known then go to 2.2 else go to 2.3

2.2 Provide answer

Add a comment to the ticket ending (if appropriate) with the sentence “If this does not fully answer your question please let us know.” Go to 2.5

2.3 Has user checked website?

If the user has already checked the website, or it is unlikely the website will have the information they are seeking, go to 2.6 else go to 2.4

2.4 Advise user to check website

Add a comment to the ticket “Extensive information about Fed4FIRE and its use is available at <http://www.fed4fire.eu/home.html> . If you are not able to find the information you need please let us know and we will investigate further for you.” Go to 2.5

2.5 [User] Question fully answered?

If the user confirms their question is answered then go to 4.1 else go to 2.3

2.6 Escalate to SME

Assign the ticket to the appropriate (based on Test-bed, or functional area). Go to 2.7

2.7 [SME] Answer question

The SME will work with whoever is required (the person asking the question, other Fed4FIRE users, other SMEs) in order to answer the question. The SME will if appropriate re-assign the ticket to other SMEs to take responsibility for. Go to 2.8

2.8 [User] Question fully answered?

If the question has been fully answered then go to 4.2 else go to 2.7

A-3) Response to Service Requests (see flowchart on p. 27)

3.1 Type of Service Request?

Check the type of service requested.

If it is for access to a Test-bed then go to 3.2

If it is a request for a new TTS user account go to 3.3

If it is any other type of Service Request go to 3.5

3.2 Escalate to SME

Assign the ticket to SME responsible for handling service requests at the given test-bed (see section "Fed4FIRE Subject Matter Experts" of this appendix). Go to 3.6

3.3 Create TTS account

FLS Create the new TTS user account using the information supplied. Go to 3.4

3.4 Resolve ticket

Resolve ticket with comment that new account has been created. Close the ticket.

3.5 Escalate to default SME

Assign the ticket to the default SME (see section "Fed4FIRE Subject Matter Experts" of this appendix) Go to 3.6

3.6 [SME] Action Request

Action the access request. Go to 3.7

3.7 [SME] Resolve Ticket

Resolve the ticket, indicating whether the request was approved or rejected. Go to 3.8

3.8 Check ticket

FLS will receive a Ticket resolution notification. Check the ticket for completeness then Close the ticket.

A-4) Resolution (see flowchart on p. 28)

4.1 Resolve the ticket

Resolve the ticket, briefly describing what was done to successfully fix the issue. Go to 4.2

4.2 KEDB update required?

If the KEDB needs updating go to 4.9 else Close the ticket.

4.3 [SME] Resolve Ticket

Resolve the ticket, briefly describing what was done to successfully fix the issue. Go to 4.5

4.4 Linked Tickets?

FLS will receive a Ticket resolution notification. Check to see if that resolved ticket has any linked tickets. If it does then for each linked ticket re-start activity at 1.5 and go to 4.5. Else go straight to 4.5

4.5 KEDB and website info update check

E-mail the SME(s) 'Please advise if any updates are required for our Known Error Database, or the Fed4FIRE website'. Add a restricted comment (viewable only by developers) to the ticket indicating this has been done.

If no response within 1 week then contact them again (but phone as well if possible). Add a restricted comment. If no response after repeated attempts then Close the ticket, with an appropriate restricted comment.

Go to 4.6

4.6 [SME] KEDB or website update required?

If KEDB needs updating go to 4.7 and/or if Fed4FIRE website needs updating go to 4.8

4.7 Provide KEDB info

Send FLS the information they need for the Known Error Database so that if this issue recurs the FLS will be able to manage the issue without escalating it to an SME. Information for the KEDB should include:

- Symptoms for the error

- Actions to be taken, and by whom

- How to determine the issue has been addressed

- If the prescribed actions do not address the issue, to whom should the issue be escalated

Go to 4.9

4.8 [SME] Update website

Provide information to webmaster so that the website may be properly updated. E-mail FLS to advise website is being updated. Go to 4.9

4.9 Update KEDB

Check any information provided by SMEs for completeness and unambiguity. Seek clarification from the SME as required. Update the KEDB with the information once it is all correct. Add a restricted comment {see section "Adding a restricted comment" in this appendix} to the ticket 'Known Error Data Base updated'. Then close the ticket.

4.10 Update ticket

Add an appropriate comment e.g. 'The Fed4FIRE website will be updated with additional guidance'. Close the ticket.

B) Service Access and Availability

The FLS Trouble Ticket System is JIRA. Fed4FIRE users (most typically ‘experimenters’) will e-mail helpdesk@fed4fire.eu in order to directly open a ticket. Users have to be registered (and with the right e-mail address) in order to open tickets. When a new Fed4FIRE user registers on the portal, the corresponding information is automatically mailed to FLS and an account on the FLS Jira is will be created. Users can also request registration via a form on the Fed4FIRE portal – this request will be mailed to Fed4FIRE FLS who will then create the account. There will also be a form on the portal that users can use to contact the FLS in the event they are having trouble sending mails to the FLS Trouble Ticket System. The system is accessible on-line at <issues.fed4fire.eu>

The DANTE Network Operations Centre (NOC) provides First Level Support (FLS) for Fed4FIRE during core working hours (0900-1700 CET).

C) Fed4FIRE Subject Matter Experts

C-1) Default SME

If it is not possible to determine, to a reasonable degree of certainty, the appropriate SME to escalate a ticket to then the ticket should be assigned to the Fed4FIRE default SME using the email address defaultsme@fed4fire.eu. At this moment this will result in assigning the ticket to the Fed4FIRE coordination team at iMinds, but this can be changed when needed in due time.

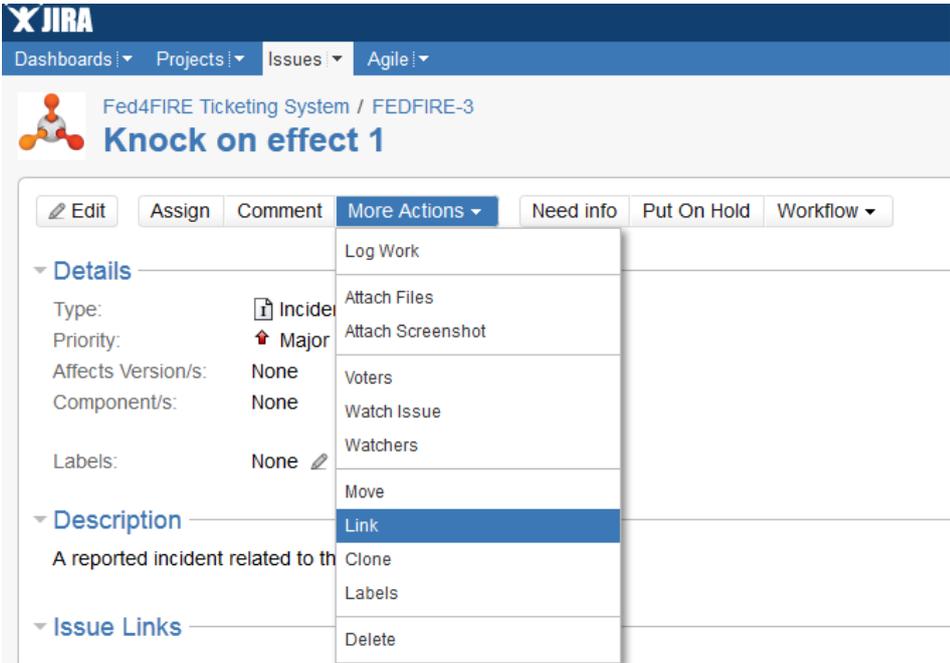
C-2) Test-bed Principle SMEs

Test-bed	Name	Contact details	Comment
Interconnection	Georgios Androulidakis	gandr@netmode.ntua.gr	
Virtual Wall	Brecht Vermeulen	brecht.vermeulen@intec.ugent.be	
w-iLab.t 2	Brecht Vermeulen	brecht.vermeulen@intec.ugent.be	
FuSeCo	Florian Schreiner	florian.schreiner@fokus.fraunhofer.de	
PlanetLab Eu.	Ciro Scognamiglio Michael Sioutis	ciro.scognamiglio@lip6.fr michael.sioutis@lip6.fr	
NITOS	Donatos Stavropoulos	dostavro@gmail.com	
NETMODE	Georgios Androulidakis	gandr@netmode.ntua.gr	
NORBIT	Max Ott	max.ott@nicta.com.au	
OFELIA	Carlos Bermudo	carlos.bermudo@i2cat.net	
KOREN	Junwoo Park	johnpa@gmail.com	

D) Examples and illustrations

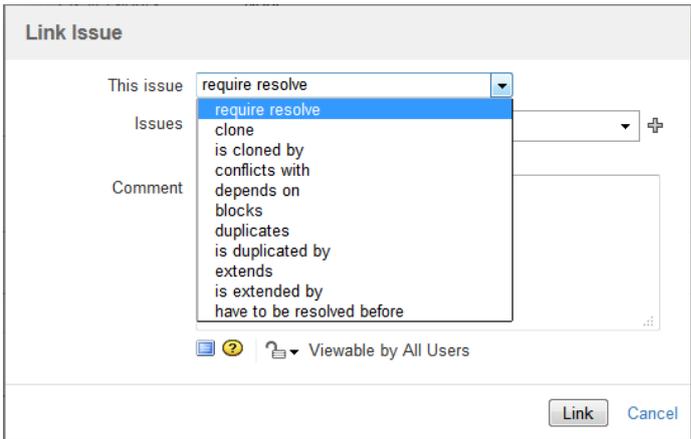
D-1) Linking JIRA Tickets

To link one JIRA ticket to another



The link type can be of various types. The most common ones are:

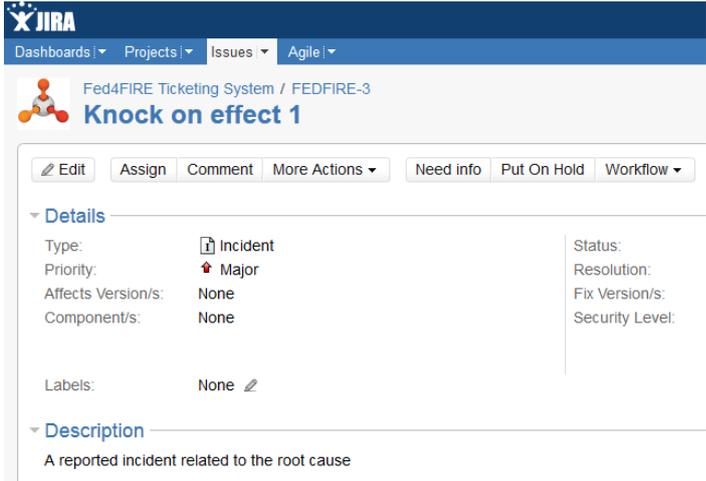
- 'Have to be resolved before' – this ticket has to be resolved before the linked ticket
- 'Require resolve' - this ticket cannot be resolved before the linked ticket



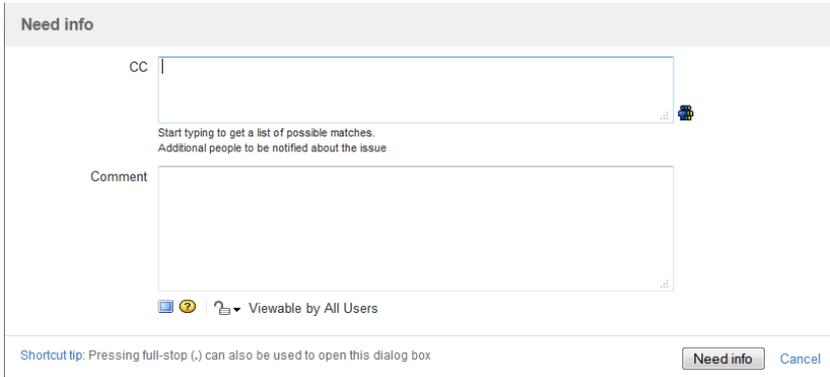
D-2) Requesting more information

To request a user to provide more info click on 'Need Info'





Then specify the info required in the pop-up comment box, adding extra people to 'cc' if required. Click on 'Need Info' to complete



D-3) Dashboard ticket summaries

Dashboard tickets have Summary titles of “<test-bed> <attribute> <condition>”. For the below example dashboard display, ticket summaries would be:

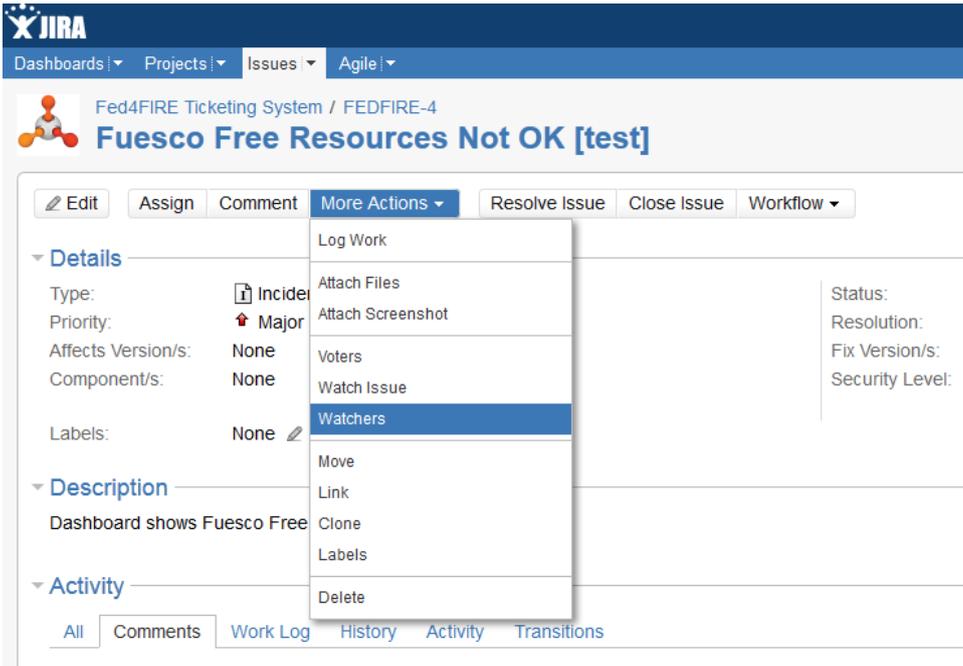
Fuesco Free Resources not OK
 NetMODE GetVersion Status not OK

Fed4FIRE First Level Support Monitoring					
Testbed Name	Ping latency (ms)	GetVersion Status	Free Resources	Internal testbed monitoring status	Last check internal status
Fuseco	23.68	ok	not ok	N/A	
NETMODE	58.63	not ok	not ok	ok	2013-06-12 12:57:22+02
Planetlab Europe	32.02	ok	301	N/A	
Virtual Wall	2.85	ok	0	ok	2013-06-12 12:53:22+02
w-ILab.t 2	3.38	ok	52	ok	2013-06-12 12:53:22+02

D-4) Add watchers to a ticket

In order to send ticket updates to additional people (other than the reporter and the assignee) then add them as 'watchers'. Such people need to have JIRA accounts with appropriate permissions – it is not possible to add just e-mail addresses.





D-5) Adding a restricted comment

By default JIRA ticket comments are visible to anyone who can access the ticket. This can be changed by mean of a dropdown list beneath the comment box.

