

Bestman Software Assessment, September 13/14 2011

Alain Roy, Brian Bockelman, John Hover, Catalin Dumescue, Doug Strain

Bestman Software Assessment, September 13/14 2011	1
Introduction, Charge.....	1
Summary.....	1
Effort Estimate	2
Action Items.....	3
Bestman Architecture	3
Bestman Code	4
Evolution of the software	5
Security.....	5
Pending Stakeholder Requests.....	5
Modules that might be deprecated.....	5
HTTPS Transition.....	5
Acknowledgements.....	6
Software Module Dependencies	6
Example Request Flow.....	7
List of files in the Bestman 2 code tree.....	7

Introduction, Charge

The main questions of the assessment group are to understand and estimate the effort and need for the support and development of the Bestman code to meet OSG Stakeholder needs.

The current Bestman team consists of three people with the following areas of responsibility/expertise.

- Alex Sim, - transfer modules, full mode
- Junmin Gu – server code,
- Vijaya Natarajan – client code

The previous level of effort available has been 0.5 FTE for support and feature addition from OSG and ~0.6 FTE from other funding sources. The other funding sources are no longer available as of September 1st 2011.

Summary

The support and development needs for Bestman are evaluated separately for the Gateway and Full modes.

The skill set needed to support and evolve Bestman includes knowledge and experience with Java and J2EE.
 Support for the Full Mode adds significantly to the effort needed for support and evolution as well as the complexity and time for testing of changes and new releases.

Effort Estimate

Based on the effort used in the existing Bestman team over the past 3 years and the initial discussion of the code base we estimate the effort for new personnel, with existing Java and J2EE skill and experience.

The estimates are with the assumption that there is no continuing support for Bestman 1 and only Bestman 2 is supported, maintained and evolved. These estimates are approximate (hopefully within a factor of 2).

Support only:

	Effort to get up to speed	Ongoing effort
Support Gateway Mode only	4 FTE months	3 FTE months/year
Support Full Mode in addition	4 FTE months	3 FTE months/year

The additional development effort needed for anticipated upgrades:

	Comments	Effort
Java 7	Depends on tests on prototype release by end of Sept 2011.	Initial build and simple test done by the assessment team works.
Https	Agree on protocol changes. Goal to meet the OSG needs and reduce the migration effort to Axis 2.0	Delegate to other SRM development projects depended on by OSG stakeholders, dCache and EOS
	Server	6 FTE months
	Client	6 FTE months
Axis 2.0	Since this breaks protocol compatability the estimate is only made if done at the same time and in conjunction with the HTTPS transition.	3 FTE months additional to https develop if done at the same time.

The assessment team also did an initial successful build and test for Scientific Linux 6. There are no outstanding OSG stakeholder requests for other OS than SL5, but there is an expectation that SL6 and Debian may happen within the next

year. Given the SL6 build the Debian build/test estimate is included in the estimated support effort above.

Action Items

Requests to the Bestman Team before end of December (Prioritized):

1. Improve the level of code comments for the code itself. As of today there are no embedded comments in the code to guide the support or developer.
2. Help full deployment of Bestman2 to all OSG installations such that the support of Bestman 1 can be dropped.
3. Determine modules with no current stakeholders that are candidates for de-support.
4. Implement better dependency management for the build.

Requests to OSG:

1. Consider whether support for opensaml can be dropped in all OSG modules (Gums and Bestman) in favor of the existing xacl implementations. This is referred to the technology group.
2. Provide a list of external Java packages relied on by Bestman to the Security Team to include in their watch for vulnerability announcements.
3. If transition of support for Bestman is to be transitioned outside of the existing team ensure 2-4 weeks of brain dump of the existing experts of someone up to speed on java and j2ee.

Bestman Architecture

The functionality of Bestman is separated into the Gateway and Full modes.

- Gateway Mode is used by all the stakeholders. The functionality of Bestman in this mode is the storage management (through the SRM interface) with data movement requested and controlled outside of Bestman by the client application directly interfacing to GridFTP.
- Full Mode is used only by the STAR experiment. The functionality includes management of the queue of data transfer requests, implemented through a database, as well as transfer of the data, response to errors etc. The Full Mode is lower performance than the Gateway Mode and does not meet the performance needs of US ATLAS, US CMS.

The Gateway mode uses the Request Queue Management and Processing modules and Disk Management, as well as one components of the Local Policy module. Full mode uses the rest of the modules.

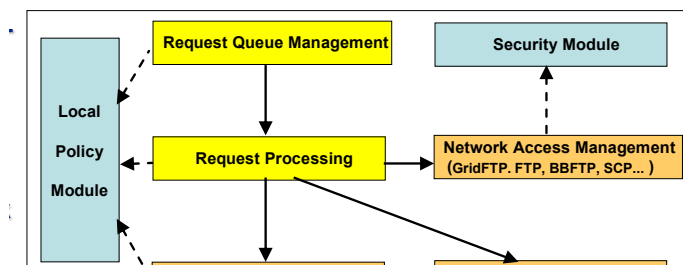


Figure 1: High level breakdown of BestMan Design

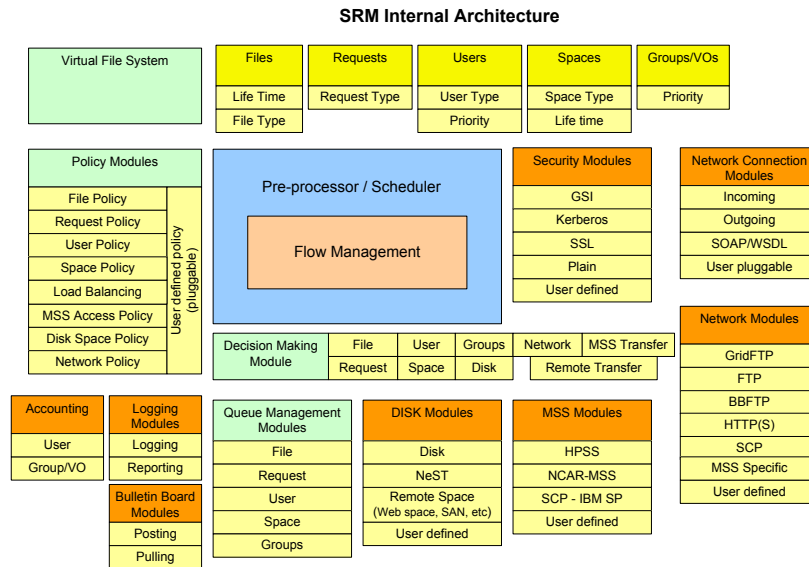


Figure 2: SRM Internal Architecture

Bestman Code

The Bestman code source is available through an open source repository maintained by LBNL as an institution. The provision of the Bestman source is currently independent of the support mechanisms. Its continued availability needs to be checked.

The Bestman code consists of ~200,000 lines of Java. About half of this is to implement the Full Mode functionality. In general, code that supports Full Mode is interlaced throughout the source modules; however, some modules are only needed for Full Mode. The client includes significant error handling code.

Subdir	# java files	# java lines
bestman2-japi-src	30	13,198
bestman2-server-src	205	64,963
bestman-client-src	144	89,601
srmtester3-src	34	31,604

Total	413	199,366

Table 1: Subdirectories of Bestman Source with count of files and lines of code

The dependencies on external packages are shown at the end of this note.

Evolution of the software

There are three major developments that are anticipated. The first 2 will result in a Bestman release that is incompatible with the current releases.

- Transition from httpg to https security infrastructure. This is being done to the protocol and time lines defined in collaboration with the EMI project and software developers of other SRM interface developer projects – dCache, Castor, DPM.
- Upgrade from Java 6 to Java 7/OpenJDK
- Upgrade from Axis 1.4 to Axis 2.0

The effort needed for each of these is listed in the summary table.

Security

We are told that there have not been any security flaws identified with the Bestman code to date. There has been development to provide access to files on disk based on individual and group rights.

Pending Stakeholder Requests

Alex told the group that there are currently no user requests or changes pending for Bestman. The experience of the team is an average of one such request monthly. These requests - either bugs or feature requests - usually require developer effort. In-house driven development effort currently in progress is the investigation more performant databases than Berekely DB to support Full Mode.

Modules that might be deprecated

At the present time there may be no stakeholders depending on the REDNET, NEST, or Bulletin Board modules.

HTTPS Transition

The transition to HTTPS is driven by the European community - specifically the EMI project. It is unknown at this time whether this is a WLCG or LHC experiment requirement.

The resulting services will be incompatible with the existing HTTPG-based releases. The need for, and agreed upon interface to, a delegation service is one of the main complexities of the transition. For Bestman the delegation service is only needed for

Full Mode. Care should be taken that the agreed upon interfaces/protocols do not break this assumption.

EMI has a goal for initial releases for dCache and DPM by the end of 2013.

Acknowledgements

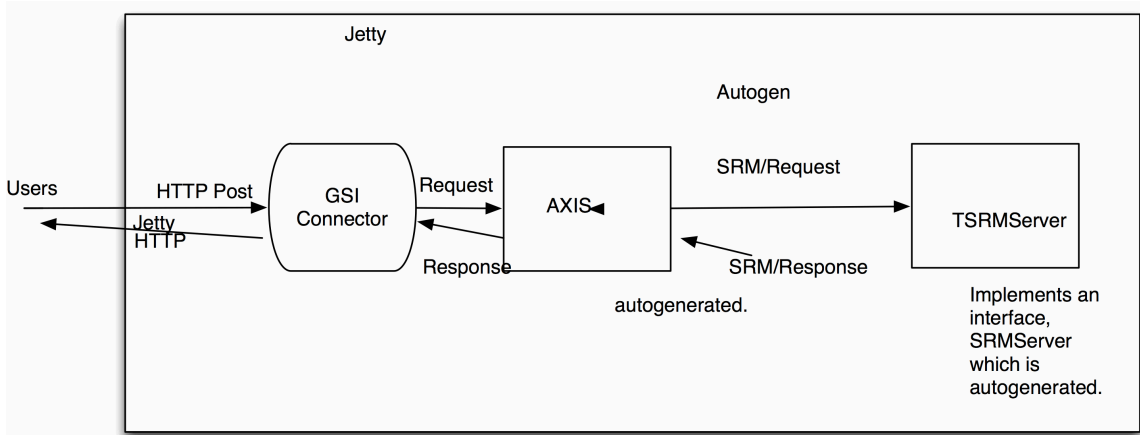
We much appreciate the presence of Alex Sim, the information and documentation he provided and his patience answering all the assessors questions. We also appreciate Andy Hanushevsky for attending at the end of the XROOTD workshop and offering whatever help is needed

Software Module Dependencies

- Oracle/SUN JDK 1.6.0_27 – Java – risk is upgrading to new JDK; licencing issue.
- Apache ant 1.7.0 – needed only for the builds
- Axis 1.4 – soap library, very old; Apache may drop it. Released in 2006. Axis 2.0. Upgrading to Axis 2 is hard and will break the interoperability between the client and the server. GUMS also uses Axis 1.4;
- WSDL4j-1.6.2 – translation of WSDL 5 to Java. For Build only.
- Xerces-2.9.1 – XML libraries, secondary dependency from Java cog and GSI. But Bestman has upgraded version
- Jetty 7.1.5.v20100705 – container for the servlets.
- Java CoG 1.8.0 – will be 2.0; GSI and GridFTP client libraries. Can drop if drop full mode.
- Java Servlet 2.5 – definition/implementation of J2EE infrastructure: code.
- Slf4j-1.6.0 - logging components. Plug/play logging infrastructure log4j would work. Provides all other library interfaces. (simple logging façade for java)
- Berkeley DB 4.0.103 – for Full mode log record functionality.
- concurrent lib 1.3.4 – handling concurrency; part of JDK 1.7; multi-threading; will be same implementation but will need to have changes in the import.
- from SUNY-Oswego
- which4j – Run time dynamic class loading. Search and for run time for the backend plug ins
- glite-security lib – VOMS and trust manager. When https version of VOMS-ADMIN comes out may not need to use Java CoG server side/GSI
- opensaml 1.0.1 – interface to GUMS, 2 versions of GUMS are supported.
- privilege 1.0.1.3 – interface to GUMS; uses opensaml 1.. If drop privilege can drop opensaml 1
- vomsjapi 1.9.10 – interface to GUMS
- commons lang 2.1 – Apache. Stable. Internal general utility. Helper utilities.
- commons collections 3.2 – Apache. Stable. Helper utilities.

- joda-time 1.5.2 – java date and time manipulation library. Not so popular as Apache.
- opensaml 2.2.1
- privilege-xacml 2.2.4 – uses opensaml 2
- velocity 1.5 – templating engine. Web service interface.
- Xalan 2.7.1 – pulled in by opensaml 2.
- XML-security 1.4.1 – GUMS xacml
- Bouncy Castle crypto lib 1.3.9 – general purpose crypto, voms, jlite trust manager.

Example Request Flow



TSRMService -> TSRMServer -> SRMRequestLS->creates (T?)SRMRequestLS (implements Renewable)

Figure 3: Example Request Flow

List of files in the Bestman 2 code tree

bestman2/lib/bestman2-client.jar
 bestman2/lib/plugin/LBNL.RFF.jar
 bestman2/lib/jglobus/cog-url-1.8.0.jar
 bestman2/lib/jglobus/jce-jdk13-131.jar
 bestman2/lib/jglobus/cog-axis-1.8.0.jar
 bestman2/lib/jglobus/puretls.jar
 bestman2/lib/jglobus/cryptix32.jar
 bestman2/lib/jglobus/cryptix-asn1.jar
 bestman2/lib/jglobus/cog-jglobus-1.8.0.jar
 bestman2/lib/jglobus/log4j-1.2.15.jar
 bestman2/lib/axis/wsdl4j-1.6.2.jar
 bestman2/lib/axis/axis.jar
 bestman2/lib/axis/commons-logging-1.1.jar
 bestman2/lib/axis/xercesImpl-2.9.1.jar
 bestman2/lib/axis/jaxrpc.jar
 bestman2/lib/axis/commons-discovery-0.2.jar

bestman2/lib/gums/glite-security-util-java.jar
bestman2/lib/gums/opensaml-1.0.1.jar
bestman2/lib/gums/privilege-1.0.1.3.jar
bestman2/lib/gums/glite-security-trustmanager.jar
bestman2/lib/gums/vomsjapi-1.9.10.jar
bestman2/lib/gums2/commons-lang-2.1.jar
bestman2/lib/gums2/joda-time-1.6.jar
bestman2/lib/gums2/xalan-2.7.1.jar
bestman2/lib/gums2/velocity-1.5.jar
bestman2/lib/gums2/opensaml-2.3.1.jar
bestman2/lib/gums2/xmltooling-1.2.1.jar
bestman2/lib/gums2/xml-apis-2.9.1.jar
bestman2/lib/gums2/privilege-xacml-2.2.5.jar
bestman2/lib/gums2/xmlsec-1.4.3.jar
bestman2/lib/gums2/openws-1.3.0.jar
bestman2/lib/gums2/commons-collections-3.2.jar
bestman2/lib/bestman2-transfer.jar
bestman2/lib/voms/bcprov-jdk15-139.jar
bestman2/lib/jetty/jetty-security-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-io-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-xml-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-http-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-util-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-deploy-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-continuation-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-webapp-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-servlet-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-client-7.1.5.v20100705.jar
bestman2/lib/jetty/jetty-server-7.1.5.v20100705.jar
bestman2/lib/others/which4j.jar