GFD-R-P.135
HPC File Staging Profile

http://forge.gridforum.org/projects/ogsa-hpcp-wg

Authors:
Glenn Wasson, UVA
Marty Humphrey, UVA

June 28, 2008

HPC File Staging Profile, Version 1.0

Status of this Memo

This memo provides information to the Grid community regarding the specification of the HPC File Staging Profile. Distribution is unlimited.

Copyright Notice

Abstract

This document profiles the File staging capabilities of the Job Submission Description Language (JSDL) for use by HPC Basic Profile-compliance services. It includes clarifications, refinements, interpretations and amplifications of JSDL which promote interoperability.

Contents

1 Introduction

The HPC File Staging Profile is a document that is used to describe an extension to the HPC Basic Profile [HPCP10]. This profile addresses how file staging can be performed by HPC Profile-compliant services using the JSDL <DataStaging> directives.

The Profile consists of references to existing specifications, along with any clarifications of the contents of those specifications, restrictions on the use of those specifications, and references to any normative extensions to those specifications. While it is envisioned that many systems will have capabilities above and beyond those described in this profile, this profile describes a basic set of capabilities that can be used as the basis for defining interoperability between clients and services claiming compliance.

The document is structured as a set of sections, each of which is used to reference a particular aspect of an HPC File Staging Profile compliant system.

2 Notational Conventions

The key words “MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,” “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” are to be interpreted as described in RFC-2119 [RFC 2119].

The document refers to an “HPC File Staging Profile compliant system” as a “Compliant system”.

This specification uses namespace prefixes throughout; they are listed in Table 2-1. Note that the choice of any namespace prefix is arbitrary and not semantically significant.

Table 2-1: Prefixes and namespaces used in this specification.

Prefix	Namespace
xsd	http://www.w3.org/2001/XMLSchema
jsdl	http://schemas.ggf.org/jsdl/2005/11/jsdl
bes-factory	http://schemas.ggf.org/bes/2006/08/bes-factory
hpcp-bp	http://schemas.ogf.org/hpcp/2007/01/bp
hpcp-fs	http://schemas.ogf.org/hpcp/2007/01/fs

3 JSDL Data Staging

This profile adopts the DataStaging elements from the Job Submission Description Language v1.0 [JSDL10]. Modifications and clarifications to those elements appear in section 3.1. In addition, the profile extends these elements by defining an additional element that may be used by clients for scheduling file transfers.

3.1 JSDL 1.0 Data Staging Elements

A system compliant with this profile MUST support the following JSDL data staging elements (with noted clarifications).

3.1.1 FileName

3.1.2 As in [JSDL10].FileSystemName

This profile does not support this element. Compliant systems which receive a JSDL document containing a <FileSystemName> element MUST return an <UnsupportedFeatureFault> fault.

3.1.3 CreationFlag

As in [JSDL10], but with the clarification that it is not considered an error if the CreationFlag is set to dontOverwrite and a file with the same name exists at the target location. The existing file is left in place and not replaced.

3.1.4 DeleteOnTerminate

As in [JSDL10]. However, this profile views the JSDL document as a contact between the user and the service. This means that any action requested by the JSDL document that cannot be completed by the service must result in an error report being sent to the client. To this end, this profile defines that failure to delete the associated file MUST move the job to the Failed state (see section 4).

3.1.5 Source

As in [JSDL10].

3.1.6 Target

As in [JSDL10].

3.2 Credentials

Files staging operations may require additional credentials in order to interact with remote systems. This profile defines an additional element, called <Credential> which can be placed in the <DataStaging> element. The value of this element is <xsd:any> and an example of this is shown below.

```
<DataStaging>
   <FileName>output.txt</FileName>
   <CreationFlag>overwrite</CreationFlag>
   <Target>
       <URI>ftp://server.inthe.sky:1234</URI>
   </Target>
   <Credential xmlns="http://schemas.ogf.org/hpcp/2007/11/ac">
       <UsernameToken xmlns="http://docs.oasis-open.org/wss/2004/01/oasis-
200401-wss-wssecurity-secext-1.0.xsd">
           <Username>demo</Username>
           <Password>pass</Password>
       </UsernameToken>
   </Credential>
</DataStaging>
```

Compliant implementations MUST recognize this element if it is present. In addition, compliant implementations MUST be able to recognize and parse at least one of the following when appearing as the Credential element's content:

- Username Token element encoding as defined by [WS-Security] and profiled by [WSSUTP] and [WS-IBSP]
- X.509 Certificate Token encoding defined in [WS-Security] and profiled by [WSSX509] and [WS-IBSP]

An implementation is free to support both of the above tokens as well as additional token types. If an implementation receives a token type that it does not recognize or support, it MUST return an <bes-factory:UnsupportedFeatureFault>.

3.3 Supported Protocols

While JSDL defines schema types for data staging elements, it does not further specify permissible values. This profile, in contrast, requires compliant systems to support a minimum set of values for those elements. Specifically, this profile requires that compliant services MUST support at least one of the following file transfer protocols: ftp [FTPRFC], http [HTTPRFC] and scp [SSHRFC]. These protocols will likely be referenced by clients using the scheme portion of the <URI> sub-elements (e.g. ftp://) within both the <Source> and <Target> elements.

In order to prevent confusion between the information contained in the <Credential> element and information contained in the URIs themselves, this profile mandates that username/password information MUST not be embedded in URIs. For example, URIs such as ftp://bob?fred@host.com/foo.txt are not allowed.

While compliant services may support the scp protocol, there is no defined standard for scp URIs. This profile defines the syntax of these URIs as being equivalent to the ftp URI syntax [RFC 3986] except that the scheme MUST be "scp://" instead of "ftp://" (and credentials may not be embedded in the URIs as noted above).

3.4 Discovery of Supported Protocols and Security Tokens

A service may wish to advertise which file transfer protocols it supports. In order to do this, we define the following URIs which identify a service as supporting the HPC-FSP as well as the supported transfer protocols. These URIs MUST be placed in the <bes-factory:BESExtension> element of the <bes-factory:FactoryResourceAttributesDocument> element.

http://www.ogf.org/hpc-fsp/2008/01/protocol/ftp identifies the service as supporting HPC-FSP using FTP

http://www.ogf.org/hpc-fsp/2008/01/protocol/http/v11 identifies the service as supporting HPC-FSP using HTTP 1.1

http://www.ogf.org/hpc-fsp/2008/01/protocol/scp identifies the service as supporting HPC-FSP using scp.

In general, URIs used to advertise a service's support for other protocols SHOULD be crafted as: http://www.ogf.org/hpc-fsp/***year***/***month***/protocol/***protocol_name[/protocol_version]*** where "year", "month", "protocol_name" and, optionally, "protocol_version" are to be filled in with appropriate values.

A service may also advertise the security tokens that it accepts for interacting with remote servers. The following URIs, placed in the <bes-factory:BESExtention> element, identify a HPC-FSP compliant service as supporting the listed security tokens.

http://www.ogf.org/hpc-fsp/2008/01/token/username identifies the service as supporting Username Tokens as defined in [WS-Security] and profiled in [WSSUTP] and [WS-IBSP]

http://www.ogf.org/hpc-fsp/2008/01/token/x509 identifies the service as supporting X.509 Certificate Tokens as defined in [WS-Security] and profiled in [WSSX509] and [WS-IBSP]

In general, URIs used to advertise a service's support for other token types SHOULD be crafted as: http://www.ogf.org/hpc-fsp/***year***/***month***/token/***token_name*** where "year", "month" and "token_name" are to be filled in with appropriate values.

3.5 File Staging Failure Semantics

A JSDL document may specify that multiple files are to be staged-in and/or staged-out. This leaves open two questions: 1) whether the job is considered to be in error if some (but not all) of

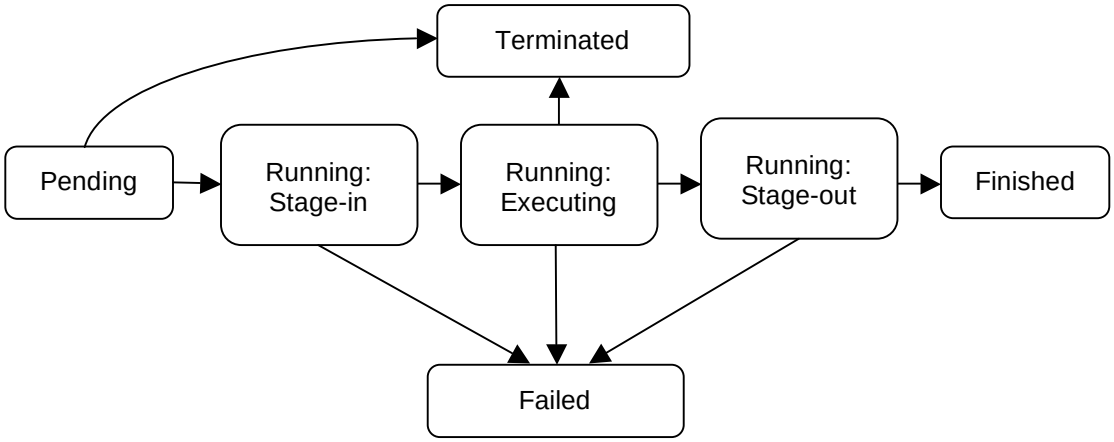
those staging requests fail and 2) how should any staging operations that are in-process or not yet started be handled once a failure occurs? In other words, should a job transition to the “failed” state as soon as any staging directive fails and should staging operations continue after a failure occurs? The later, in particular, may have different answers for the stage-in and stage-out cases. For stage-in, it may be reasonable to stop staging once a file transfer fails (under the assumption that the job will not be able to run without all of its input data available), while for stage-out it may be reasonable to attempt to stage-out every requested file even if some fail.

Since undoubtly services complying with this profile will be based on infrastructures with diverse failure models/semantics, this profile places no mandate on the semantics of failure in multiple file staging operations. A service MAY transition to the failed state when a file staging failure occurs. In addition, a service MAY abort current and/or uninitiated transfers when a staging failure occurs. Finally, a service MAY support extensions to this profile by which clients can specify a particular set of desirable semantics.

4 Service State Model

This profile extends the HPC Base Profile, which uses the BES state model, to include new states for stage-in and stage-out file transfer. These states are sub-states of the Running state.

The new state diagram is shown below.



These new states are defined as follows:

- Running:Stage-in – file transfer operations are underway, but the application specified by the JSDL document has not yet begun executing
- Running:Executing – the application is currently executing
- Running:Stage-out – file transfer operations are underway and the application is no longer executing

A service compliant with this profile MAY support these sub-states, meaning that the service may respond to GetActivityStatuses requests with these sub-states. If a service does support these sub-states, it MUST respond using the elements <hpcp-fs:Stage-in />, <hpcp-fs:Executing /> and <hpcp-fs:Stage-out/>. An example response message for the Running:Stage-in state is shown below (other sub-states are specified by replacing the <Running:Stage-in> element with one of the other elements shown above).

```
<bes-factory:GetActivityStatusesResponse>
   <bes-factory:ActivityIdentifier>
      <wsa:Address>http://tempuri.org/some-service</wsa:Address>
      <wsa:ReferenceParameters>
         <n00:id>D4A88953-FFFF-49F6-5145-AE21FF0438AE</n00:id>
      </wsa:ReferenceParameters>
   </bes-factory:ActivityIdentifier>
   <bes-factory:ActivityStatus>
      <bes-factory:State>Running</bes-factory:State>
      <hpcp-fs:Stage-in />
   </bes-factory:ActivityStatus>
</bes-factory:GetActivityStatusesResponse>
```

4.1 Reporting File Staging Failures

While the above state model provides distinct state transitions for failures related to staging-in, staging-out and executing, these transitions may occur for many different reasons. In order to more precisely describe the source of failure to clients, this profile extends the ActivityStatusType defined in BES [BES10] to include fault information (using the built-in extensibility of that element). An example message is shown below:

```
<bes-factory:GetActivityStatusesResponse>
   <bes-factory:ActivityIdentifier>
      <wsa:Address>http://tempuri.org/some-service</wsa:Address>
      <wsa:ReferenceParameters>
         <n00:id>D4A88953-FFFF-49F6-5145-AE21FF0438AE</n00:id>
      </wsa:ReferenceParameters>
   </bes-factory:ActivityIdentifier>
   <bes-factory:ActivityStatus>
      <bes-factory:State>Failed</bes-factory:State>
      <soap:Fault>
         <soap:faultcode> some code </soap:FaultCode>
         <soap:detail> a fault description, e.g. a fault schema from section 4.2 </>
      </soap:Fault>
   </bes-factory:ActivityStatus>
</bes-factory:GetActivityStatusesResponse>
```

A SOAP 1.1 fault has been added to the <bes-factory:ActivityStatus> element. This fault could describe the source of the stage-in failure, e.g. insufficient disk space or unknown source file. Note that this fault is different from the Fault element which may be present as a sub-element of the <bes-factory:GetActivityStatusesResponse> element. While the later indicates that a fault occurred in querying the status of an activity, the former provides details useful to determining why the activity is in its current failure state.

4.2 File Staging Faults

We define the following standard fault types that can be placed in the <detail> element of the SOAP faults introduced in section 4.1. It should be noted that determining the cause of a file staging failure can be difficult particularly when the error occurs on a remote system such as when performing a stage-out operation. As such, these error messages SHOULD be used when an appropriate cause for the failure can be determined, but it is not mandatory to throw one of these faults.

FileNotFoundFault – this fault SHOULD be thrown when the file indicated by the <Source> element or the remote server/directory indicated by the <Target> element cannot be found.

```
<hpcp-fs:FileNotFoundFault>
   xsd:string
</hpcp-fs:FileNotFoundFault>
```

UnsupportedProtocolFault – this fault SHOULD be thrown when the file indicated by the <Source> or <Target> element specifies a transfer protocol that is not supported by the service. The <File> sub-element indicates the file who's URI contains the unsupported protocol.

```
<hpcp-fs:UnsupportedProtocolFault>
   xsd:string
</hpcp-fs:UnsupportedProtocolFault>
```

NotAuthorizedFault – this fault, defined in BES 1.0 [BES10], SHOULD be thrown when the stage-in or stage-out requests cannot be completed due to insufficient permissions of the entity performing the staging operation.

DeleteOnTerminationFault – this fault SHOULD be thrown if a staging request has the DeleteOnTermination flag set to true and the service is unable to delete the specified file. The <File> sub-element indicates the file while was not deleted.

```
<hpcp-fs:DeleteOnTerminationFault>
   xsd:string
</hpcp-fs:DeleteOnTerminationFault>
```

LocalStagingFault – this fault SHOULD be thrown when an error occurs during file staging that is not covered by one of the previously defined faults and that failure occurred on the local system, e.g. a file was transferred from a remote server, but failed on write to local disk. The value of this fault element may contain additional text to describe the failure.

```
<hpcp-fs:LocalStagingFault>
   xsd:string
</hpcp-fs:LocalStagingFault>
```

RemoteStagingFault – this fault SHOULD be thrown when a failure occurs during file staging that is not covered by one of the previously defined faults and that failure occurred on a remote server, e.g. the remote server closed the connection. The value of this fault element may contain additional text to describe the failure.

```
<hpcp-fs:RemoteStagingFault>
   xsd:string
</hpcp-fs:RemoteStagingFault>
```

5 Author Information

Glenn Wasson
University of Virginia

Marty Humphrey
University of Virginia

6 Contributors & Acknowledgements

We gratefully acknowledge the contributions made to this specification by Steven Newhouse, Chris Smith, Vesso Novov, Bill Nitzberg, Blair Dillaway, and Donal Fellows.

We would like to thank the people who took the time to read and comment on earlier drafts. Their comments were valuable in helping us improve the readability and accuracy of this document.

7 Full Copyright Notice

8 Intellectual Property Statement

The OGF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the OGF Secretariat.

The OGF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this recommendation. Please address the information to the OGF Executive Director.

Disclaimer

This document and the information contained herein is provided on an “As Is” basis and the OGF disclaims all warranties, express or implied, including but not limited to any warranty that the use of the information herein will not infringe any rights or any implied warranties of merchantability or fitness for a particular purpose.

9 Normative References

[RFC 2119] Bradner, S. *Key words for use in RFCs to Indicate Requirement Levels*. Internet Engineering Task Force, RFC 2119, March 1997. Available at http://www.ietf.org/rfc/rfc2119.txt

[JSDL10] Anjomshoaa, A., Brisard, F., Drescher, M., Fellows, D., Ly, A., McGough, S., Pulsipher, D. and Savva, A. *Job Submission Description Language (JSDL) Specification, Version 1.0*. GFD 56. Available at http://www.ggf.org/documents/GFD.56.pdf.

[BES10] Foster, I., Grimshaw, A., Lane, P., Lee, W., Morgan, M., Newhouse, S., Pickles, S. Pulsipher, D., Smith, C., and Theimer, M. *OGSA Basic Execution Service, Version 1.0*. GFD 108. August 2007. Available at http://www.ogf.org/documents/GFD.108.pdf.

[HPCP10] Dillaway, B., Humphrey, M., Smith, C., Theimer, M. and Wasson, G. *HPC Basic Profile, Version 1.0*. GFD 114. August 2007. Available at http://www.ogf.org/documents/GFD.114.pdf.

[WSSUTP] Nadalin, A., Griffin, P., Kaler, C., Hallam-Baker, P., and Monzillo, R. eds. *Web Services Security UsernameToken Profile 1.0*. March 2004. Available at http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-username-token-profile-1.0.pdf.

[WSSX509] Hallam-Baker, P., Kaler, C., Monzillo, R., Nadalin, A. eds. *Web Services Security X.509 Certificate Token Profile 1.0*. March 2004. Available at http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-token-profile-1.0.pdf.
[RFC 3986] Berners-Lee, T., Fielding, R. and Masinter, L. *Uniform Resource Identifier (URI): Generic Syntax.* Internet Engineering Task Force, RFC 3986, January 2005. Available at: http://www.ietf.org/rfc/rfc3986.txt
[WS-Security] Nadalin, A., Kaler, C., Monzillo, R. and Hallam-Baker, P. eds. *Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)*. February 2006. Available at http://docs.oasis-open.org/wss/v1.1/
[FTPRFC] Postel, J. and Reynolds, J. *File Transfer Protocol (FTP)*. October 1985. Available at: http://www.faqs.org/rfcs/rfc959.html

[HTTPRFC] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and Berners-Lee, T. *Hypertext Transfer Protocol – HTTP/1.1*. June 1999. Available at: http://www.w3.org/Protocols/rfc2616/rfc2616.html

[SSHRFC] Ylonen, T. and Lonvick, C. *The Secure Shell (SSH) Protocol Architecture*. January 2006. Available at http://www.ietf.org/rfc/rfc4251.txt.