GFD-R-P.XXX DRMAA-WG drmaa-wg@ogf.org Daniel Gruber, Univa September 2014

Distributed Resource Management Application API Version 2 (DRMAA) Go Language Binding

Status of This Document

OGF Proposed Recommendation (GFD-R-P.XX)

Document Change History

Date	Notes
September XXth, 2014	Submission to OGF Editor
October XXth, 2014	Updates from public comment period
November XXth, 2014	Publication as GFD-R-P.XXX

Copyright Notice

Copyright © Open Grid Forum (2014). Some Rights Reserved. Distribution is unlimited.

Trademark

All company, product or service names referenced in this document are used for identification purposes only and may be trademarks of their respective owners.

Abstract

This document describes the Go language binding for the Distributed Resource Management Application API Version 2 (DRMAA). The intended audience for this specification are DRMAA implementors.

¹Corresponding author

Notational Conventions

In this document, C language elements and definitions are represented in a fixed-width font.

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 \cite{black} ?].

Contents

1	Introduction	4
2	General Design	4
	2.1 Error Handling	4
	2.2 Lists and Dictionaries	4
3	Implementation-specific Extensions	Ç
4	Go Public Interface Documentation	Ę
5	Contributors	14
6	Intellectual Property Statement	14
7	Disclaimer	14
8	Full Copyright Notice	14
a	References	1 6

1 Introduction

The Distributed Resource Management Application API Version 2 (DRMAA) specification defines an interface for tightly coupled, but still portable access to the majority of DRM systems. The scope is limited to job submission, job control, reservation management, and retrieval of job and machine monitoring information.

The *DRMAA root specification* [?] describes the abstract API concepts and the behavioral rules of a compliant implementation, while this document standardizes the representation of API concepts in the Go programming language.

2 General Design

The mapping of DRMAA IDL constructs to Go follows a set of design principles. Implementation-specific extensions of the DRMAA Go API SHOULD follow these conventions:

- Name spaces of the DRMAA API, as demanded by by the root specification, is realized with the drmaa2_ Go package name.
- The methods are implemented on structs representing the object name from the IDL specification.
- Return types are references to allocated structs.
- The implementation MAY be based on the C language binding.
- Go comes with its own garbage collector. The implementation SHOULD avoids to call wrappers for the functions defined in the C standard for freeing the data. This usually requires that the methods transforms and copies data available in the C API into its own Go specific counterparts.
- Nevertheless for complete memory management a destructor or an exit function needs to be called.
- Functions are returning additionally an Go type error value. This MAY be casted by the calling function to a drmaa2. Error type which encodes the error reason as a constant, as well as a human readable string representation. The Go implementation MUST provide an return value which fulfills this request (see Error Handling section).
- Go default types are going to be used, i.e. Go maps for dictionaries and slices for lists of return values.
- UNSET values are derived from the underlying C DRMAA values.
- Time values which define a duration are represented with the Go time. Duration type. An UNSET duration is XXX.

2.1 Error Handling

Methods are returning the Go error type. If no error happened then nil is returned otherwise the error is set. The error MUST be able to be casted to the drmaa2.Error type which is a pointer to a struct consisting of a constant indicating the error reason as well as a string value which describes the error in human readable form. Both values SHOULD provide the same information as the C API calls to drmaa2_lasterror and drmaa2_lasterror_text (if they are called directly after the function call in the same thread).

2.2 Lists and Dictionaries

Lists are in the Go binding slices while Dictionaries are implemented as Go maps.

DRMAA interface	Go binding prefix
DrmaaReflective	struct methods (Go embedding) see below
SessionManager	SessionManager struct
JobSession	JobSession struct
ReservationSession	ReservationSession struct
MonitoringSession	MonitoringSession struct
Reservation	Reservation struct
Job	Job struct
JobArray	ArrayJob struct
JobTemplate	JobTemplate struct
ReservationTemplate	ReservationTemplate struct

Table 1: Mapping of DRMAA interface name to Go structs on which methods are defined

3 Implementation-specific Extensions

The DRMAA root specification allows the product-specific extension of the DRMAA API in a standardized way.

Go method available for an (implementation specific) extensible data structure	Description
GetExtension(key) (string, error)	Returns an value of an implementation specific extension.
SetExtension(key, value string) error DescribeExtension(string) (string, error)	Sets an extension to a specific value. Returns a string containing a human readable description of the extension.
ListExtensions() []string	Lists all available extension keys for the specific object.

Table 2: Methods available for all extensible DRMAA data structures.

4 Go Public Interface Documentation

The following text shows the complete Go interface description generated by the go doc? tool.

DRMAA-compliant Go libraries MUST declare all functions and data structures described here. Implementations MAY add custom parts in adherence to the extensibility principles of this specification and the root specification.

The source file is also available at http://www.drmaa.org.

```
package drmaa2
    import "github.com/dgruber/drmaa2"

CONSTANTS

const (
    JobTemplateType = iota
    JobInfoType
    ReservationTemplateType
```

```
ReservationInfoType
        QueueInfoType
        MachineInfoType
        NotificationType
const (
        AdvanceReservation = iota
        ReserveSlots
        Callback
        BulkJobsMaxParallel
        JtEmail
        JtStaging
        JtDeadline
        JtMaxSlots
        JtAccountingId
        RtStartNow
        RtDuration
        RtMachineOS
        RtMachineArch
const (
        NewState = iota
        {\tt Migrated}
        AttributeChange
const InfiniteTime = int64(C.DRMAA2_INFINITE_TIME)
    Special timeout value: Wait probably infinitly
const ZeroTime = int64(C.DRMAA2_ZERO_TIME)
        Special timeout value: Don't _{\sqcup}\,\text{wait}
TYPES
\verb"type" \verb| ArrayJob" \verb| struct" \{
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{contains} {\scriptstyle \sqcup} \texttt{filtered} {\scriptstyle \sqcup} \texttt{or} {\scriptstyle \sqcup} \texttt{unexported} {\scriptstyle \sqcup} \texttt{fields}
\texttt{func}_{\,\sqcup\,}(\texttt{aj}_{\,\sqcup\,} * \texttt{ArrayJob})_{\,\sqcup\,} \texttt{GetId}()_{\,\sqcup\,} \texttt{string}
\verb| u u u u Returns | uthe | uidentifier | uof | uthe | uArray Job .
\texttt{func}_{\,\sqcup\,}(\texttt{aj}_{\,\sqcup\,}*\texttt{ArrayJob})_{\,\sqcup\,}\texttt{GetJobTemplate}\,()_{\,\sqcup\,}*\texttt{JobTemplate}
\verb| u u u u | Returns u the u JobTemplate u of u an u Array Job .
\texttt{func}_{\,\sqcup\,}(\texttt{aj}_{\,\sqcup\,}*\texttt{ArrayJob})_{\,\sqcup\,}\texttt{GetJobs}()_{\,\sqcup\,}[]\,\,\texttt{Job}
\verb""" Returns" \verb""" a \verb""" list" of \verb""" individual" jobs" the \verb""" Array Job" consists" of .
\texttt{func}_{\,\sqcup\,}(\texttt{aj}_{\,\sqcup\,}*\texttt{ArrayJob})_{\,\sqcup\,}\texttt{GetSessionName}()_{\,\sqcup\,}\texttt{string}
\verb| u u u u Returns | | the uname | of uthe | | job | | session | | the uarray | | job | | belongs | | to |.
\texttt{func}_{\,\sqcup\,}(\texttt{aj}_{\,\sqcup\,}*\texttt{ArrayJob})_{\,\sqcup\,}\texttt{Hold}()_{\,\sqcup\,}\texttt{error}
UUUUSetsuallutasksuofuanuArrayJobutouhold.
func_(aj_*ArrayJob)_Release()_error
func_(aj_*ArrayJob)_Resume()_error
\verb| uuuu Resumes uallususpended utasks uof uan u Array Job.
funcu(aju*ArrayJob)uSuspend()uerror
\verb"uuuu" Suspends" \verb"all" running" \verb"tasks" of \verb"uan" Array Job.
funcu(aju*ArrayJob)uTerminate()uerror
\label{eq:local_substitution} $$ _{\sqcup \sqcup \sqcup \sqcup}$ Terminates_{\sqcup}(usually_{\sqcup}sends_{\sqcup}a_{\sqcup}SIGKILL)_{\sqcup}all_{\sqcup}tasks_{\sqcup}of_{\sqcup}an_{\sqcup}ArrayJob \,.
\texttt{type}_{\,\sqcup\,} \mathtt{CPU}_{\,\sqcup\,} \mathtt{int}
{}^{-}_{\sqcup \sqcup \sqcup \sqcup} \mathtt{CPU}_{\sqcup} \mathtt{architecture}_{\sqcup} \mathtt{types}
const<sub>11</sub>(
\sqcup \sqcup \sqcup \sqcup \sqcup \mathsf{OtherCPU} \sqcup \mathsf{CPU} \sqcup = \sqcup \mathsf{iota}
_{\sqcup \sqcup \sqcup \sqcup} \mathtt{Alpha}
UUUU ARM
```

```
⊔⊔⊔⊔ ARM64
uuuu Cell
UUUU PA_RISC
UUUU PA_RISC64
UUUUIA_64
UUUUMIPS
⊔⊔⊔⊔MIPS64
⊔⊔⊔⊔PowerPC
⊔⊔⊔⊔PowerPC64
UUUU SPARC
SPARC64
\texttt{func}_{\,\sqcup\,}(\texttt{cpu}_{\,\sqcup\,}\texttt{CPU})_{\,\sqcup\,}\texttt{String}\,()_{\,\sqcup\,}\texttt{string}
type \, {}_{\sqcup} \, CallbackFunction \, {}_{\sqcup} \, func \, (notification \, {}_{\sqcup} \, Notification)
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} A {\scriptstyle \sqcup} \operatorname{Callback} {\scriptstyle \sqcup \operatorname{is} \sqcup \operatorname{a} \sqcup} \operatorname{function} {\scriptstyle \sqcup} \operatorname{which} {\scriptstyle \sqcup} \operatorname{works} {\scriptstyle \sqcup \operatorname{on} \sqcup} \operatorname{the} {\scriptstyle \sqcup} \operatorname{notification} {\scriptstyle \sqcup} \operatorname{struct}.
\texttt{type}_{\,\sqcup\,} \texttt{Capability}_{\,\sqcup\,} \texttt{int}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup} Capabilities {\scriptstyle \sqcup} are {\scriptstyle \sqcup} optional {\scriptstyle \sqcup} functional ities {\scriptstyle \sqcup} defined {\scriptstyle \sqcup} by {\scriptstyle \sqcup} the {\scriptstyle \sqcup} DRMAA2
_{\sqcup\sqcup\sqcup\sqcup}standard.
\verb"type" \verb| Drmaa2Extensible" interface" \{
{}_{\sqcup \sqcup \sqcup \sqcup} / / {}_{\sqcup} Lists {}_{\sqcup} all {}_{\sqcup} implementation {}_{\sqcup} specific {}_{\sqcup} key {}_{\sqcup} names {}_{\sqcup} for
_{\sqcup \sqcup \sqcup \sqcup} ListExtensions()_{\sqcup}[]string
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathsf{DescribeExtension} \, (\, \mathsf{string}) \, {\scriptstyle \sqcup} \, \mathsf{string} \,
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{SetExtension}(\texttt{string})_{\sqcup} \texttt{error}
GetExtension()ustring }
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} The_{\sqcup} Drmaa 2 Extensible_{\sqcup} interface_{\sqcup} lists_{\sqcup} all_{\sqcup} functions_{\sqcup} required_{\sqcup} for_{\sqcup} DRMAA 2 all_{\sqcup} functions_{\sqcup} required_{\sqcup} for_{\sqcup} functions_{\sqcup} required_{\sqcup} for_{\sqcup} functions_{\sqcup} functions_{\sqcup} required_{\sqcup} for_{\sqcup} functions_{\sqcup} fun
\verb|uuuu| extensible | | data | | structures | | (JobTemplate, | UJobInfo | | etc.)|.
 \texttt{type}_{\,\sqcup} \texttt{Error}_{\,\sqcup} \, \texttt{struct}_{\,\sqcup} \{
_{\text{\tiny $\square\square\square\square$}} \texttt{Message}_{\text{\tiny $\square$}} \texttt{string}
{\scriptstyle \square \square \square \square} {\scriptstyle \square} {\scriptstyle Id} {\scriptstyle \square \square \square \square \square} {\scriptstyle \square} {\scriptstyle \square} {\scriptstyle ErrorId}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, DRMAA2 {\scriptstyle \sqcup} \, error {\scriptstyle \sqcup} \, (\, implements {\scriptstyle \sqcup} \, GO {\scriptstyle \sqcup} \, Error {\scriptstyle \sqcup} \, interface \, ) \, .
\texttt{func}_{\,\sqcup\,}(\texttt{ce}_{\,\sqcup\,}\texttt{Error})_{\,\sqcup\,}\texttt{Error}()_{\,\sqcup\,}\texttt{string}
\verb| U U U U U The | U DRMAA2 | Error | implements | Uthe | Error | interface.
\texttt{func}_{\,\sqcup\,}(\texttt{ce}_{\,\sqcup\,}\texttt{Error})_{\,\sqcup\,}\texttt{String}\,()_{\,\sqcup\,}\texttt{string}
\verb| uuuu Implement| \verb| the | \verb| Stringer| \verb| interface | \verb| for | \verb| an | \verb| drmaa2. Error
type_{\sqcup}ErrorId_{\sqcup}int
UUUUDRMAA2uerroruID
____Success_ErrorId_=_iota
\sqcup \sqcup \sqcup \sqcup \sqcup DeniedByDrms
UUUUU DrmCommunication
⊔⊔⊔⊔TryLater
UUUUU Session Management
_{\sqcup \sqcup \sqcup \sqcup}Timeout
_{\cup\cup\cup\cup\cup}Internal
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt InvalidArgument}
⊔⊔⊔⊔InvalidSession
_{\sqcup\sqcup\sqcup\sqcup}InvalidState
⊔⊔⊔⊔ OutOfResource
{\scriptscriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt UnsupportedAttribute}
UnsupportedOperation
____ImplementationSpecific
LastError
type_{\sqcup}Event_{\sqcup}int
\texttt{type}_{\,\sqcup\,} \texttt{Extension}_{\,\sqcup\,} \texttt{struct}_{\,\sqcup\,} \{
\tt UUUUUSType_{UUUUUUU}StructType_{UUUUUUUU}//_UStores_{U}the_{U}type_{U}of_{U}the_{U}struct
{\scriptstyle \text{UUUU}} Internal_{\text{UUUUUU}} unsafe \,.\, Pointer_{\text{UUUU}} //{\scriptstyle \text{U}} Enhancmement_{\text{U}} of_{\text{U}} C_{\text{U}} struct
```

```
\verb|uu|u| ExtensionListumap[string] string| // ustores uthe uextension urequests uas ustring | vertex under the uextension urequests used to the under the u
 \verb| uuuu Extension | \verb| struct | \verb| which | \verb| is | \verb| embedded | \verb| in | DRMAA2 | \verb| objects | \verb| which | \verb| are | are
 UUUU extensible.
 \texttt{func}_{\,\sqcup\,}(\texttt{e}_{\,\sqcup\,}*\texttt{Extension})_{\,\sqcup\,}\texttt{GetExtension}(\texttt{extension}_{\,\sqcup\,}\texttt{string})_{\,\sqcup\,}(\texttt{string}\,,_{\,\sqcup\,}\texttt{error})
 \verb| uuuu| For uall utypes u which u embedds uthe u Extension ustruct u (JobTemplate u etc.) 
  type,,Job,,struct,,{
 _{\sqcup \sqcup \sqcup \sqcup \sqcup} / / _{\sqcup} contains _{\sqcup} filtered _{\sqcup} or _{\sqcup} unexported _{\sqcup} fields _{\sqcup}
 func_{\sqcup}(job_{\sqcup}*Job)_{\sqcup}GetId()_{\sqcup}string
 funcu(jobu*Job)uGetJobInfo()u(*JobInfo,uerror)
 \verb"uuu" information" about" the \verb"ujob".
 func_{\sqcup}(job_{\sqcup}*Job)_{\sqcup}GetJobTemplate()_{\sqcup}(*JobTemplate,_{\sqcup}error)
 \verb| u u u u | Returns | u the | u JobTemplate | u u sed | u to | u submit | u the | u job .
 func_(job_*Job)_GetSessionName()_string
 func_{\sqcup}(job_{\sqcup}*Job)_{\sqcup}GetState()_{\sqcup}JobState
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} the {\scriptstyle \sqcup} current {\scriptstyle \sqcup} JobState {\scriptstyle \sqcup} of {\scriptstyle \sqcup} the {\scriptstyle \sqcup} job \,.
 \texttt{func}_{\sqcup}(\texttt{job}_{\sqcup}*\texttt{Job})_{\sqcup}\texttt{Hold}()_{\sqcup}\texttt{error}
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup} Puts {\scriptstyle \sqcup} the {\scriptstyle \sqcup} job {\scriptstyle \sqcup} into {\scriptstyle \sqcup} an {\scriptstyle \sqcup} hold {\scriptstyle \sqcup} state {\scriptstyle \sqcup} so {\scriptstyle \sqcup} that {\scriptstyle \sqcup} it {\scriptstyle \sqcup} is {\scriptstyle \sqcup} not {\scriptstyle \sqcup} scheduled. {\scriptstyle \sqcup} If {\scriptstyle \sqcup} the {\scriptstyle \sqcup} job {\scriptstyle \sqcup} into {\scriptstyle \sqcup} scheduled.
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \mathsf{is} \sqcup \mathsf{already} \sqcup \mathsf{running} \sqcup \mathsf{it} \sqcup \mathsf{continues} \sqcup \mathsf{to} \sqcup \mathsf{run} \sqcup \mathsf{and} \sqcup \mathsf{the} \sqcup \mathsf{hold} \sqcup \mathsf{state} \sqcup \mathsf{becomes} \sqcup \mathsf{only}}
 \verb| u u u u effectice | when | uthe | ujob | uis | ure scheduled.
 \texttt{func}_{\sqcup}(\texttt{job}_{\sqcup}*\texttt{Job})_{\sqcup}\texttt{Release}()_{\sqcup}\texttt{error}
 \verb|uu|u| Releases | the | job | from | hold | state | so | that | it | will | be | schedulable.
 func_{\sqcup}(job_{\sqcup}*Job)_{\sqcup}Resume()_{\sqcup}error
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Resumes {\scriptstyle \sqcup} a {\scriptstyle \sqcup} job {\scriptstyle \sqcup} / {\scriptstyle \sqcup} process {\scriptstyle \sqcup} to {\scriptstyle \sqcup} be {\scriptstyle \sqcup} runnable {\scriptstyle \sqcup} again \, .
 \texttt{func}_{\,\sqcup\,}(\,\texttt{job}_{\,\sqcup}\,\texttt{*}\,\texttt{Job}\,)_{\,\sqcup\,}\texttt{Suspend}\,(\,)_{\,\sqcup\,}\texttt{error}
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Stops {\scriptstyle \sqcup} a {\scriptstyle \sqcup} job {\scriptstyle \sqcup} / {\scriptstyle \sqcup} process {\scriptstyle \sqcup} from {\scriptstyle \sqcup} beeing {\scriptstyle \sqcup} executed \,.
 \texttt{func}_{\,\sqcup\,}(\,\texttt{job}_{\,\sqcup\,}*\,\texttt{Job})_{\,\sqcup\,}\texttt{WaitStarted}\,(\,\texttt{timeout}_{\,\sqcup\,}\texttt{int}\,\texttt{64}\,)_{\,\sqcup\,}\texttt{error}
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup} Blocking \sqcup wait \sqcup until \sqcup the \sqcup job \sqcup is \sqcup started . \sqcup The \sqcup timeout \sqcup prefents \sqcup that \sqcup the \sqcup
 \verb| u u u u call u is u blocking u endlessly . \verb| u Special u timeouts u are u available u by u the all the state of the 
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, constants \, {\scriptstyle \sqcup} \, Infinite \\ Time \, {\scriptstyle \sqcup} \, and \, {\scriptstyle \sqcup} \, Zero \\ Time \, .
  func_{\sqcup}(job_{\sqcup}*Job)_{\sqcup}WaitTerminated(timeout_{\sqcup}int64)_{\sqcup}error
 \verb| uuuu| Waits| until| the | job| goes| into| one| of| the | finished| states. | The| timeout
 {}_{\sqcup \sqcup \sqcup \sqcup} \texttt{specifies} {}_{\sqcup} \texttt{the} {}_{\sqcup} \texttt{maximum} {}_{\sqcup} \texttt{time} {}_{\sqcup} \texttt{to} {}_{\sqcup} \texttt{wait} \cdot {}_{\sqcup} \texttt{If} {}_{\sqcup} \texttt{no} {}_{\sqcup} \texttt{timeout} {}_{\sqcup} \texttt{is} {}_{\sqcup} \texttt{required} {}_{\sqcup} \texttt{use} {}_{\sqcup} \texttt{the} {}_{\sqcup} \texttt{the} {}_{\sqcup} \texttt{model} \mathsf{the} \mathsf
 \sqcup \sqcup \sqcup \sqcup \sqcup constant \sqcup InfiniteTime.
  UUUUU//UreferenceUtoUtheUvoid*UpointerUwhich
 UUUUU//uisuuseduforuextensions
 \square\square\square\square\squareExtension
 UUUUIduuuuuuuuuuustring
 UUUU ExitStatus UUUUUUU int
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{TerminatingSignal} {\scriptstyle \sqcup} \texttt{string}
 {\scriptstyle \text{UUUU}} \, \texttt{Annotation} \\ {\scriptstyle \text{UUUU}} \, \texttt{string}
 {\scriptstyle \text{\tiny UUUU}} State_{\scriptstyle \text{\tiny UUUUUUUUUUU}} JobState
 UUUU SubState UUUUUUUUU string
 ⊔⊔⊔⊔ AllocatedMachines [] string
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Submission Machine {\scriptstyle \sqcup} string
 {\scriptstyle \text{uuuu}} \, \texttt{JobOwner}_{\text{uuuuuuuuu}} \, \texttt{string}
 {\scriptstyle \text{\tiny UUUU}} \texttt{Slots} {\scriptstyle \text{\tiny UUUU}} {\scriptstyle \text{\tiny UUUU}} \texttt{int} 64
 {\scriptstyle \text{\tiny UUUU}}\, Queue \texttt{Name}_{\text{\tiny UUUUUUUU}} \texttt{string}
 WallclockTime UUUUUtime. Duration
UUUU CPUTimeuuuuuuuuuuint64
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \mathtt{DispatchTime}_{\sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} \mathtt{time} \, . \, \mathtt{Time}
-------time.Time.Time.Time.Time.
```

```
funcu(structTypeu*JobInfo)uListExtensions()u[]string
 {}^{-}_{\sqcup \sqcup \sqcup \sqcup} Returns_{\sqcup} a_{\sqcup} string_{\sqcup} list_{\sqcup} containing_{\sqcup} all_{\sqcup} implementation_{\sqcup} specific_{\sqcup} extensions
 UUUUUofutheu JobInfou object.
   func_{\sqcup}(ji_{\sqcup}*JobInfo)_{\sqcup}SetExtension(extension,_{\sqcup}value_{\sqcup}string)_{\sqcup}error
   \texttt{type}\,{\sqcup}\, \texttt{JobSession}\,{\sqcup}\, \texttt{struct}\,{\sqcup} \{
 {\tt \_}{\tt \_}{\tt \_}{\tt \_}{\tt Name}{\tt \_}{\tt string}{\tt \_}//{\tt \_}{\tt public}{\tt \_}{\tt name}{\tt \_}{\tt of}{\tt \_}{\tt job}{\tt \_}{\tt session}
funcu(jsu*JobSession)uClose()uerror
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, {\tt Closes} \, {\scriptstyle \sqcup} \, {\tt an} \, {\scriptstyle \sqcup} \, {\tt open} \, {\scriptstyle \sqcup} \, {\tt JobSession} \, .
 func_{\sqcup}(js_{\sqcup}*JobSession)_{\sqcup}GetContact()_{\sqcup}(string,_{\sqcup}error)
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} the {\scriptstyle \sqcup} contact {\scriptstyle \sqcup} string {\scriptstyle \sqcup} of {\scriptstyle \sqcup} the {\scriptstyle \sqcup} DRM {\scriptstyle \sqcup} session \,.
 \texttt{func}_{\sqcup}(\texttt{js}_{\sqcup}*\texttt{JobSession})_{\sqcup}\texttt{GetJobArray}(\texttt{id}_{\sqcup}\texttt{string})_{\sqcup}(*\texttt{ArrayJob}\,,_{\sqcup}\texttt{error})
 \verb|_u| \verb|_u| \verb|_Returns_u| \verb|_a| reference_u| to_u| \verb|_an_u| existing_u| \verb|_Array| Job_u| based_u| on_u| the_u| given_u| job_u| id.
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} In_{\sqcup} case_{\sqcup} of_{\sqcup} an_{\sqcup} error_{\sqcup} the_{\sqcup} error_{\sqcup} return_{\sqcup} value_{\sqcup} is_{\sqcup} set_{\sqcup} to_{\sqcup} ! =_{\sqcup} nil \,.
 \texttt{func}_{\sqcup}(\texttt{js}_{\sqcup}*\texttt{JobSession})_{\sqcup}\texttt{GetJobCategories}(\texttt{)}_{\sqcup}(\texttt{[]}\texttt{string}\,,_{\sqcup}\texttt{error})
 _{\sqcup \sqcup \sqcup \sqcup} Returns_{\sqcup} all_{\sqcup} job_{\sqcup} categories_{\sqcup} specified_{\sqcup} for_{\sqcup} the_{\sqcup} job_{\sqcup} session.
   func_{\sqcup}(js_{\sqcup}*JobSession)_{\sqcup}GetJobs()_{\sqcup}([]Job,_{\sqcup}error)
 {}_{\sqcup \sqcup \sqcup \sqcup} Returns {}_{\sqcup} a_{\sqcup} list {}_{\sqcup} of {}_{\sqcup} all {}_{\sqcup} jobs {}_{\sqcup} currently {}_{\sqcup} running {}_{\sqcup} in {}_{\sqcup} the {}_{\sqcup} given {}_{\sqcup} JobS ession .
   \texttt{func}_{\sqcup}(\texttt{js}_{\sqcup}*\texttt{JobSession})_{\sqcup}\texttt{GetSessionName}()_{\sqcup}(\texttt{string},_{\sqcup}\texttt{error})
  func_{\sqcup}(js_{\sqcup}*JobSession)_{\sqcup}RunBulkJobs(jt_{\sqcup}JobTemplate,_{\sqcup}begin_{\sqcup}int,_{\sqcup}end_{\sqcup}int,_{\sqcup}step_{\sqcup}int,_{\sqcup}maxParallel_{\sqcup}int)_{\sqcup}(*ArrayJob,_{\sqcup}error) 
 {}_{\sqcup \sqcup \sqcup \sqcup} Submits {}_{\sqcup} a_{\sqcup} JobTemplate {}_{\sqcup} to {}_{\sqcup} the {}_{\sqcup} cluster {}_{\sqcup} as {}_{\sqcup} array {}_{\sqcup} job {}_{\sqcup} (multiple {}_{\sqcup} instances) to {}_{\sqcup} the {}_{\sqcup} th
 \verb| u| \verb| u| \verb| u| \verb| of \verb| u| the \verb| u| same \verb| u| job , \verb| u| not \verb| u| neccessarly \verb| u| running \verb| u| a \verb| u| the \verb| u| same \verb| u| point \verb| u| in \verb| u| time) . \verb| u| It means the contraction of the 
 {}_{\sqcup \sqcup \sqcup \sqcup} requires {}_{\sqcup} a_{\sqcup} Job Template {}_{\sqcup} filled {}_{\sqcup} out {}_{\sqcup} at {}_{\sqcup} least {}_{\sqcup} with {}_{\sqcup} a_{\sqcup} Remote Command . {}_{\sqcup} The
 \verb|uuuu| begin, \verb|uendu| and \verb|ustepu| parameters \verb|uspecifyingu| how \verb|umanyu| array \verb|ujobu| instances|
 _{\sqcup \sqcup \sqcup \sqcup} are_{\sqcup} \, submitted_{\sqcup} and_{\sqcup} how_{\sqcup} the_{\sqcup} \, instances_{\sqcup} are_{\sqcup} numbered_{\sqcup} \, (1 \, , 10 \, , 1_{\sqcup} denotes_{\sqcup} 10 \, , 1_{\sqcup} 1
 \verb|uu|u| = \texttt{array} | \texttt{job}| | \texttt{instances}| | \texttt{numbered}| | \texttt{from}| | \texttt{1}| | \texttt{to}| | \texttt{10}). \\ | \texttt{The}| | \texttt{maxParallel}| | \texttt{parameter}| | \texttt{paramete
 \verb|u|| \verb|u|| \verb|specifies|| \verb|how|| \verb|many|| of || the || array|| job|| instances|| should|| run|| at || parallel|| as || the ||
 \verb| u| \verb| u| \verb| u| \verb| maximum | \verb| (when | | resources | | are | | contrainted | | then | | less | | instances | | could | | run).
   \texttt{func}_{\sqcup}(\texttt{js}_{\sqcup}*\texttt{JobSession})_{\sqcup}\texttt{RunJob}(\texttt{jt}_{\sqcup}\texttt{JobTemplate})_{\sqcup}(*\texttt{Job},_{\sqcup}\texttt{error})
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup} Submits {\scriptstyle \sqcup} a {\scriptstyle \sqcup} job {\scriptstyle \sqcup} based {\scriptstyle \sqcup} on {\scriptstyle \sqcup} the {\scriptstyle \sqcup} parameters {\scriptstyle \sqcup} specified {\scriptstyle \sqcup} in {\scriptstyle \sqcup} the {\scriptstyle \sqcup} Job Template {\scriptstyle \sqcup} in
 \verb|uu| \verb|uu| the | \verb|cluster.| \verb|u| In | \verb|case| of | \verb|usuccess| | \verb|it|| returns | \verb|a|| pointer| | to | \verb|a|| Job| element, | to | \verb|a|| for the last of the l
 {}_{\sqcup \sqcup \sqcup \sqcup} which_{\sqcup} can_{\sqcup} be_{\sqcup} used_{\sqcup} for_{\sqcup} further_{\sqcup} processing ._{\sqcup} In_{\sqcup} case_{\sqcup} of_{\sqcup} an_{\sqcup} error_{\sqcup} the_{\sqcup} erro
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{return} {\scriptstyle \sqcup} \texttt{value} {\scriptstyle \sqcup} \texttt{is} {\scriptstyle \sqcup} \texttt{set} \, .
   \verb|func|| (js| * JobSession)| | \verb|WaitAnyStarted(jobs|| [] Job, | | timeout|| int64)| | (* Job, | | error) |
 \verb| uuuuu Waits| until uany u of uthe ugiven ujobs uis ustarted u (usually uin urunning ustate).
 \verb|uu| = \texttt{ven} | \texttt{when} | \texttt{none} | \texttt{of} | \texttt{the} | \texttt{given} | \texttt{jobs} | \texttt{was} | \texttt{started} | \texttt{USpecial} | \texttt{timeout} | \texttt{values} | \texttt{are} | \texttt{values} | \texttt{v
 {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \texttt{InfiniteTime} \, {\scriptstyle \sqcup} \, \texttt{and} \, {\scriptstyle \sqcup} \, \texttt{ZeroTime} \, .
   funcu(jsu*JobSession)uWaitAnyTerminated(jobsu[]Job,utimeoutuint64)u(*Job,uerror)
 \verb|u|| \verb|Waits|| \verb|until|| \verb|any|| of || the || given || jobs|| is || finished. || The || timeout || determines || the || timeout |
 \verb|uu|u| \verb|given|u| \verb|jobs|u| is u finished . \verb|uSepecial|u| timeout u| values u| are u Infinite Time u| and u| timeout u| values u| timeout u| t
 ⊔⊔⊔⊔ZeroTime.
   type_{\sqcup}JobState_{\sqcup}int
 uuuu Jobu States
 const<sub>u</sub>(
 UUUUU Undetermined U JobState U = Uiota
 _{\sqcup \sqcup \sqcup \sqcup} Queued
 ⊔⊔⊔⊔QueuedHeld
 _{\sqcup \sqcup \sqcup \sqcup} Running
 ⊔⊔⊔⊔Suspended
 ⊔⊔⊔⊔Requeued
 ⊔⊔⊔⊔RequeuedHeld
 Done
 _{\cup\cup\cup\cup\cup}Failed
```

```
func_{\sqcup}(js_{\sqcup}JobState)_{\sqcup}String()_{\sqcup}string
uuuu Implementsutheu Stringeruinterface
type_JobTemplate_struct_{{}}{} {}
LUULExtension
{\scriptstyle \square \square \square \square} Remote Command {\scriptstyle \square \square \square \square \square} string
UUUUArgsuuuuuuuuuu[]string
UUUU SubmitAsHolduuuuu bool
LUUU ReRunnable LUUUUUU bool
JobEnvironment Lulumap [string] string
⊔⊔⊔⊔WorkingDirectory⊔⊔string
UUUU JobCategory UUUUUU string
UUUU Emailuuuuuuuuuu [] string
{\scriptstyle \sqcup \sqcup \sqcup \sqcup} {\tt EmailOnTerminated}_{\sqcup} {\tt bool}
{\scriptstyle \text{\tiny UUUU}} \\ \texttt{JobName} \\ {\scriptstyle \text{\tiny UUUUUUUUU}} \\ \texttt{string}
UUUU InputPathuuuuuuuustring
{\scriptstyle \text{UUUU}} \, \texttt{OutputPath}_{\text{UUUUUUUU}} \, \texttt{string}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt ErrorPath} {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt string}
{\scriptstyle \text{\tiny UUUU}} JoinFiles_{\scriptstyle \text{\tiny UUUUUUUU}} bool
{\scriptstyle \sqcup \sqcup \sqcup \sqcup} \mathtt{ReservationId}_{\sqcup \sqcup \sqcup \sqcup \sqcup} \mathtt{string}
UUUU QueueNameuuuuuuuustring
{\scriptstyle \text{UUUU}}\texttt{MinSlots}_{\text{UUUUUUUU}}\texttt{int}64
UUUU MaxSlotsuuuuuuuuu int64
{\scriptstyle \text{\tiny UUUU}} Priority_{\scriptstyle \text{\tiny UUUU}} {\scriptstyle \text{\tiny UUUU}} int 64
_{\sqcup \sqcup \sqcup \sqcup} CandidateMachines_{\sqcup} [] string
{\scriptstyle \text{UUUU}} \texttt{MinPhysMemory} {\scriptstyle \text{UUUUU}} \texttt{int} 64
{\scriptstyle \text{\tiny UUUU}} \texttt{MachineOs} {\scriptstyle \text{\tiny UUUUUUUU}} \texttt{string}
{\scriptstyle \text{UUUU}}\texttt{MachineArch}_{\text{UUUUUUU}}\texttt{string}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \mathtt{StartTime}_{\sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} \mathtt{time} \, . \, \mathtt{Time}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathsf{DeadlineTime}_{\, \sqcup \, \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} \mathsf{time} \, . \, \mathsf{Time} \,
\verb| uuuu StageInFiles | uuuuu map [string] string|
\verb| u u u u StageOutFiles | u u u u u map [string] string|
\verb| u u u u Resource Limits | u u u u map [string] string|
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} A \texttt{ccountingId} {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{string}
\texttt{func}_{\,\sqcup\,}(\texttt{jt}_{\,\sqcup\,}*\texttt{JobTemplate})_{\,\sqcup\,}\texttt{DescribeExtension}(\texttt{extensionName}_{\,\sqcup\,}\texttt{string})_{\,\sqcup\,}(\texttt{string}\,,_{\,\sqcup\,}\texttt{error})
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} the {\scriptstyle \sqcup} description {\scriptstyle \sqcup} of {\scriptstyle \sqcup} an {\scriptstyle \sqcup} implementation {\scriptstyle \sqcup} specific {\scriptstyle \sqcup} JobTemplate
\verb"uuuu" extension" \verb"as" \verb"au" string".
\texttt{func}_{\,\sqcup\,}(\texttt{structType}_{\,\sqcup\,}*\texttt{JobTemplate})_{\,\sqcup\,}\texttt{ListExtensions}()_{\,\sqcup\,}[]\,\texttt{string}
{\tt uuuu} Returns {\tt uau} string {\tt ulist} {\tt u} containing {\tt uall} {\tt uimplementation} {\tt uspecific} {\tt uextensions}
uuuuofutheu JobTemplateu object.
\texttt{func}_{\,\sqcup\,}(\,\texttt{jt}_{\,\sqcup\,}*\,\texttt{JobTemplate}\,)_{\,\sqcup\,}\texttt{SetExtension}\,(\,\texttt{extension}\,,\,_{\,\sqcup\,}\texttt{value}_{\,\sqcup\,}\texttt{string}\,)_{\,\sqcup\,}\texttt{error}
\texttt{type} \, {\sqcup} \, \texttt{Machine} \, {\sqcup} \, \texttt{struct} \, {\sqcup} \, \{
⊔⊔⊔⊔ Extension
UUUU Nameuuuuuuuuustring
UUUU Availableuuuuu bool
UUUU Socketsuuuuuuu int64
□□□□□CoresPerSocket□int64
\square ThreadsPerCore \square int 64
{\scriptstyle \cup \cup \cup \cup} Load_{\cup \cup \cup \cup \cup \cup \cup \cup \cup \cup \cup} float 64
Physical Memory int 64
UUUUU Virtual Memory UU int 64
⊔⊔⊔⊔Architecture⊔⊔⊔CPU
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathsf{OSVersion} {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathsf{Version}
____OS___OS
____OS____OS
}
\texttt{func}_{\sqcup}(\texttt{structType}_{\sqcup} * \texttt{Machine})_{\sqcup} \texttt{ListExtensions}()_{\sqcup}[] \, \texttt{string}
\texttt{func}_{\,\sqcup\,}(\texttt{m}_{\,\sqcup\,}*\texttt{Machine})_{\,\sqcup\,} \texttt{SetExtension}(\texttt{extension}\,,_{\,\sqcup\,} \texttt{value}_{\,\sqcup\,} \texttt{string})_{\,\sqcup\,} \texttt{error}
\verb"type" \verb| | \verb"MonitoringSession" \verb| | struct" | \{
```

```
_{\sqcup \sqcup \sqcup \sqcup \sqcup} / /_{\sqcup} contains _{\sqcup} filtered _{\sqcup} or _{\sqcup} unexported _{\sqcup} fields }
func_{\sqcup}(ms_{\sqcup}*MonitoringSession)_{\sqcup}CloseMonitoringSession()_{\sqcup}error
\texttt{func}_{\,\sqcup\,}(\texttt{ms}_{\,\sqcup\,}*\texttt{MonitoringSession})_{\,\sqcup\,}\texttt{GetAllJobs}()_{\,\sqcup\,}(\texttt{jobs}_{\,\sqcup\,}[]\,\texttt{Job}\,,_{\,\sqcup\,}\texttt{err}_{\,\sqcup\,}\texttt{error})
\verb"uuuu" Returns" \verb"au" slice" of \verb"ujobs" \verb"currently" visible" \verb"in" the \verb"monitoring" session.
{\tt func_{\sqcup}(ms_{\sqcup}*MonitoringSession)_{\sqcup}GetAllMachines()_{\sqcup}(machines_{\sqcup}[]Machine,_{\sqcup}err_{\sqcup}error)}
{}_{\sqcup \sqcup \sqcup \sqcup} Returns {}_{\sqcup} a {}_{\sqcup} list {}_{\sqcup} of {}_{\sqcup} all {}_{\sqcup} machines {}_{\sqcup} configured {}_{\sqcup} in {}_{\sqcup} cluster.
\texttt{func}_{\sqcup}(\texttt{ms}_{\sqcup} * \texttt{MonitoringSession})_{\sqcup} \texttt{GetAllQueues}()_{\sqcup}(\texttt{queues}_{\sqcup}[] \texttt{Queue},_{\sqcup} \texttt{err}_{\sqcup} \texttt{error})
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} all {\scriptstyle \sqcup} queues {\scriptstyle \sqcup} configured {\scriptstyle \sqcup} in {\scriptstyle \sqcup} the {\scriptstyle \sqcup} cluster \,.
\texttt{type}\,{\scriptstyle \sqcup}\, \texttt{Notification}\,{\scriptstyle \sqcup}\, \texttt{struct}\,{\scriptstyle \sqcup}\, \{
uuuu Evtuuuuuuu Event
{\scriptstyle \text{\tiny UUUU}} \textbf{JobId}_{\text{\tiny UUUUUUU}} \textbf{string}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt SessionName}_{\sqcup} {\tt string}
_____Juneustring
_____JobState
}
\texttt{type}_{\sqcup} \texttt{OS}_{\sqcup} \texttt{int}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \mathtt{Operating} {\scriptstyle \sqcup} \mathtt{System} {\scriptstyle \sqcup} \mathtt{type}
const<sub>u</sub>(
\sqcup \sqcup \sqcup \sqcup \sqcup 0ther0S \sqcup 0S \sqcup = \sqcupiota
_{\text{\tiny $\square$}\,\text{\tiny $\square$}\,\text{\tiny $\square$}\,\text{\tiny $\square$}\,\text{\tiny $\square$}}\,\text{AIX}
_{\sqcup \sqcup \sqcup \sqcup} \mathtt{BSD}
\sqcup \sqcup \sqcup \sqcup \sqcup Linux
_{\sqcup \sqcup \sqcup \sqcup} \mathtt{HPUX}
\sqcup \sqcup \sqcup \sqcup \sqcup \mathtt{IRIX}
_{\text{\tiny UUUU}}\text{MacOS}
_{\sqcup \sqcup \sqcup \sqcup} \mathtt{SunOS}
⊔⊔⊔⊔ TRU64
_{\sqcup \sqcup \sqcup \sqcup} {\tt UnixWare}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup} \, \mathbb{V} \, \text{in}
{\scriptscriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, {\tt WinNT}
\texttt{func}_{\,\sqcup\,}(\,\texttt{os}_{\,\sqcup\,}\texttt{OS})_{\,\sqcup\,}\texttt{String}\,(\,)_{\,\sqcup\,}\texttt{string}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, An_{\sqcup} \, OS_{\sqcup} \, struct_{\sqcup} needs_{\sqcup} to_{\sqcup} be_{\sqcup} printable \, .
type | Queue | struct | {
⊔⊔⊔⊔Extension
{\tt \_\_\_\_Name\_string}
func_{\sqcup}(structType_{\sqcup}*Queue)_{\sqcup}ListExtensions()_{\sqcup}[]string
{\tt uuuuReturnsuau} string {\tt ulistu} containing {\tt ualluimplementationus} pecific {\tt uextensions}
UUUUofutheu Queueuobject.
\texttt{func}_{\,\sqcup\,}(\mathtt{q}_{\sqcup}*\mathtt{Queue})_{\,\sqcup\,}\mathtt{SetExtension}\,(\mathtt{extension}\,,{}_{\sqcup}\mathtt{value}_{\sqcup}\mathtt{string})_{\,\sqcup\,}\mathtt{error}
{\tt type}_{\sqcup}{\tt Reservation}_{\sqcup}{\tt struct}_{\sqcup}\{
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt SessionName} {\scriptstyle \sqcup \sqcup \sqcup \sqcup} {\tt String}
UUUUU Contact UUUUUUU String
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Template {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} Reservation Template
ReservationIdustring
\texttt{func}_{\,\sqcup\,}(\texttt{r}_{\,\sqcup\,}*\texttt{Reservation})_{\,\sqcup\,}\texttt{GetId}()_{\,\sqcup\,}(\texttt{string}\,,_{\,\sqcup\,}\texttt{error})
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} the {\scriptstyle \sqcup} advance {\scriptstyle \sqcup} reservation {\scriptstyle \sqcup} id
func_\(\((r_\) * Reservation)_\(\)GetInfo()_\(\)(*ReservationInfo,\(\)error)
\verb| uuuuu | Returns | uthe | ure servation | uinfo | uobject | uof | uthe | ure servation
func_{\sqcup}(r_{\sqcup}*Reservation)_{\sqcup}GetSessionName()_{\sqcup}(string,_{\sqcup}error)
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} the {\scriptstyle \sqcup} name {\scriptstyle \sqcup} of {\scriptstyle \sqcup} the {\scriptstyle \sqcup} reservation
```

```
func_{\sqcup}(r_{\sqcup}*Reservation)_{\sqcup}GetTemplate()_{\sqcup}(*ReservationTemplate,_{\sqcup}error)
func_{\sqcup}(r_{\sqcup}*Reservation)_{\sqcup}Terminate()_{\sqcup}error
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt Cancels} {\scriptstyle \sqcup} {\tt an} {\scriptstyle \sqcup} {\tt advance} {\scriptstyle \sqcup} {\tt reservation}
type_{\,\sqcup\,} Reservation Info_{\,\sqcup\,} struct_{\,\sqcup\,} \{
ReservationId
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Reservation {\tt Name} {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} string
□□□□ ReservationStartTime time. Time
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Reservation End Time {\scriptstyle \sqcup \sqcup \sqcup \sqcup} time \ . \ Time
\verb""" \texttt{ACL}" \texttt{UUUU} \texttt{ACL}" \texttt{UUUU} \texttt{UUUU} \texttt{UUUU} \texttt{UUU} \texttt{I]} \texttt{string}
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Reserved Slots {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} int 64
type,ReservationSession,struct,{
_____/_contains_filtered_or_unexported_fields }
func_{\sqcup}(rs_{\sqcup}*ReservationSession)_{\sqcup}Close()_{\sqcup}error
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Closes {\scriptstyle \sqcup} an {\scriptstyle \sqcup} open {\scriptstyle \sqcup} Reservation Session \,.
\texttt{func}_{\,\sqcup\,}(\texttt{rs}_{\,\sqcup\,}*\texttt{ReservationSession})_{\,\sqcup\,}\texttt{GetContact}()_{\,\sqcup\,}(\texttt{string}\,,_{\,\sqcup\,}\texttt{error})
\verb| u u u u u | Returns | uthe | u contact | ustring | uof | uthe | ure servation | usession.
\texttt{func}_{\sqcup}(\texttt{rs}_{\sqcup} * \texttt{ReservationSession})_{\sqcup} \texttt{GetReservation}(\texttt{rid}_{\sqcup} \texttt{string})_{\sqcup}(* \texttt{Reservation},_{\sqcup} \texttt{error})
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} a {\scriptstyle \sqcup} reservation {\scriptstyle \sqcup} object {\scriptstyle \sqcup} based {\scriptstyle \sqcup} on {\scriptstyle \sqcup} the {\scriptstyle \sqcup} AR {\scriptstyle \sqcup} id
\verb|func_{\sqcup}(rs_{\sqcup}*ReservationSession)_{\sqcup}GetReservations()_{\sqcup}([]Reservation,_{\sqcup}error)|
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} a {\scriptstyle \sqcup} list {\scriptstyle \sqcup} of {\scriptstyle \sqcup} available {\scriptstyle \sqcup} advance {\scriptstyle \sqcup} reservations
\texttt{func}_{\sqcup}(\texttt{rs}_{\sqcup}*\texttt{ReservationSession})_{\sqcup}\texttt{GetSessionName}()_{\sqcup}(\texttt{string}\,,_{\sqcup}\texttt{error})
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} the {\scriptstyle \sqcup} name {\scriptstyle \sqcup} of {\scriptstyle \sqcup} the {\scriptstyle \sqcup} reservation {\scriptstyle \sqcup} session
\texttt{func}_{\sqcup}(\texttt{rs}_{\sqcup} * \texttt{ReservationSession})_{\sqcup} \texttt{RequestReservation}(\texttt{rtemplate}_{\sqcup} \texttt{ReservationTemplate})_{\sqcup}(* \texttt{Reservation},_{\sqcup} \texttt{error})
\verb"\uu_{uuuu}Allocates" an \verb"\uadvance" reservation" based \verb"\uadvance" the \verb"\uadvance" reservation" based \verb"\uadvance" and uadvance" and uadvance and uadva
 \verb|type| LeservationTemplate| Struct| \{
{\scriptscriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} {\tt Extension}
UUUU Nameuuuuuuuuuuuustring
UUUUUStartTimeuuuuuuuutime.Time
{\scriptstyle \cup \cup \cup \cup} EndTime {\scriptstyle \cup \cup \cup \cup \cup \cup \cup \cup \cup \cup} time \,. \, Time
{\scriptstyle \text{UUUU}} \, Duration \\ {\scriptstyle \text{UUUU}} \, Uuuuuuuutime \, . \, Duration
UUUUU MinSlotsuuuuuuuuint64
UUUUU MaxSlotsuuuuuuuuuu int64
UUUUU JobCategory UUUUUUUU string
UUUUU UsersACLuuuuuuuuu [] string
UUUUU CandidateMachines [] string
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{MinPhysMemory} {\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{int} 64
UUUUU MachineOsuuuuuuuuuuustring
____string uuuu Machine Archuuuuuu string }
{\tt type}_{\sqcup} {\tt SessionManager}_{\sqcup} {\tt struct}_{\sqcup} \{
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup \sqcup} A_{\sqcup} Create {\scriptstyle \sqcup} Method {\scriptstyle \sqcup} which {\scriptstyle \sqcup} initializes {\scriptstyle \sqcup} the {\scriptstyle \sqcup} values {\scriptstyle \sqcup} and {\scriptstyle \sqcup} also {\scriptstyle \sqcup} does
uuuuinitializationuaboutucapabilities,uversionsuetc.u?!?
\verb|func_{\sqcup}(sm_{\sqcup}*SessionManager)_{\sqcup}CreateJobSession(sessionName,_{\sqcup}contact_{\sqcup}string)_{\sqcup}(*JobSession,_{\sqcup}error)|
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Creates {\scriptstyle \sqcup} a {\scriptstyle \sqcup} new {\scriptstyle \sqcup} persistent {\scriptstyle \sqcup} job {\scriptstyle \sqcup} session {\scriptstyle \sqcup} and {\scriptstyle \sqcup} opens {\scriptstyle \sqcup} it \,.
func_{\sqcup}(sm_{\sqcup}*SessionManager)_{\sqcup}CreateReservationSession(sessionName,_{\sqcup}contact_{\sqcup}string)_{\sqcup}(rs_{\sqcup}*ReservationSession,_{\sqcup}err_{\sqcup}error)
\verb| u u u u u Creates | u u Reservation Session | by u name | u and u contact | u string.
\texttt{func}_{\,\sqcup\,}(\texttt{sm}_{\,\sqcup\,}*\texttt{SessionManager})_{\,\sqcup\,}\texttt{DestroyJobSession}(\texttt{sessionName}_{\,\sqcup\,}\texttt{string})_{\,\sqcup\,}\texttt{error}
UUUUU Destroysuaujobusessionubyuname.
 func_{\cup}(sm_{\cup}*SessionManager)_{\cup}DestroyReservationSession(sessionName_{\cup}string)_{\cup}error
```

```
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathsf{Destroys} \, {\scriptstyle \sqcup} \, \mathsf{a} \, {\scriptstyle \sqcup} \, \mathsf{reservation} \, {\scriptstyle \sqcup} \, \mathsf{by} \, {\scriptstyle \sqcup} \, \mathsf{name} \, .
\texttt{func}_{\sqcup}(\texttt{sm}_{\sqcup} * \texttt{SessionManager})_{\sqcup} \texttt{GetDrmsName}()_{\sqcup}(\texttt{string},_{\sqcup} \texttt{error})
UUUUReturnsutheunameuofutheuDistributeduResourceuManagementuSystem.
\texttt{func}_{\sqcup}(\texttt{sm}_{\sqcup} * \texttt{SessionManager})_{\sqcup} \texttt{GetDrmsVersion}()_{\sqcup}(* \texttt{Version},_{\sqcup} \texttt{error})
\verb| Luu | Returns | the | version | of | the | Distributed | Resource | Management | System.
funcu(smu*SessionManager)uGetJobSessionNames()u([]string,uerror)
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} Returns {\scriptstyle \sqcup} all {\scriptstyle \sqcup} job {\scriptstyle \sqcup} sessions {\scriptstyle \sqcup} accessable {\scriptstyle \sqcup} to {\scriptstyle \sqcup} the {\scriptstyle \sqcup} user \ .
\texttt{func}_{\sqcup}(\texttt{sm}_{\sqcup} * \texttt{SessionManager})_{\sqcup} \texttt{GetReservationSessionNames}()_{\sqcup}([] \texttt{string},_{\sqcup} \texttt{error})
\_\_\_\_
func_{\sqcup}(sm_{\sqcup}*SessionManager)_{\sqcup} OpenJobSession(sessionName_{\sqcup}string)_{\sqcup}(js_{\sqcup}*JobSession,_{\sqcup}err_{\sqcup}error)
\verb|func|| (sm| * Session Manager)| | Open Monitoring Session (session Name|| string)| | (* Monitoring Session , || error)| | (* Monitoring Session , || error)
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \mathsf{Opens} {\scriptstyle \sqcup} \mathsf{a} {\scriptstyle \sqcup} \mathsf{MonitoringSession} {\scriptstyle \sqcup} \mathsf{by} {\scriptstyle \sqcup} \mathsf{name} \, . \\ {\scriptstyle \sqcup} \mathsf{Usually} {\scriptstyle \sqcup} \mathsf{the} {\scriptstyle \sqcup} \mathsf{name} \, {\scriptstyle \sqcup} \mathsf{ignored} \, .
\verb|func|| (sm|| *SessionManager)| | OpenReservationSession (name|| string)| | (rs|| ReservationSession , || err|| error) | (rs|| ReservationSession , || err|| error , || err|| error , || err
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathtt{Opens} \, {\scriptstyle \sqcup} \, \mathtt{an} \, {\scriptstyle \sqcup} \, \mathtt{existing} \, {\scriptstyle \sqcup} \, \mathtt{ReservationSession} \, {\scriptstyle \sqcup} \, \mathtt{by} \, {\scriptstyle \sqcup} \, \mathtt{name} \, .
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \, \mathtt{whenever} \, {\scriptstyle \sqcup} \, \mathtt{a} \, {\scriptstyle \sqcup} \, \mathtt{event} \, {\scriptstyle \sqcup} \, \mathtt{occured} \, .
\texttt{func}_{\,\sqcup\,}(\texttt{sm}_{\,\sqcup\,}*\texttt{SessionManager})_{\,\sqcup\,}\texttt{Supports}(\texttt{c}_{\,\sqcup\,}\texttt{Capability})_{\,\sqcup\,}\texttt{bool}
\verb| u u u u Checks | whether | uthe | DRMAA2 | uimplementation | supports | an | optional
\texttt{type}_{\sqcup} \texttt{StructType}_{\sqcup} \texttt{int}
\verb| In_{\sqcup} order_{\sqcup} to_{\sqcup} make_{\sqcup} extension_{\sqcup} functions_{\sqcup} dependend_{\sqcup} from_{\sqcup} the_{\sqcup} type_{\sqcup} of_{\sqcup} the
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{struct} {\scriptstyle \sqcup} \texttt{we} {\scriptstyle \sqcup} \texttt{need} {\scriptstyle \sqcup} \texttt{to} {\scriptstyle \sqcup} \texttt{store} {\scriptstyle \sqcup} \texttt{the} {\scriptstyle \sqcup} \texttt{type} {\scriptstyle \sqcup} \texttt{somewhere} \, .
type | Version | struct | {
{\scriptstyle \sqcup \sqcup \sqcup \sqcup \sqcup} \texttt{Major}{\scriptstyle \sqcup \texttt{String}}
____Minor_string
func (v → * Version) String() string
```

5 Contributors

Daniel Gruber

Univa GmbH c/o Rüter und Partner Prielmayerstr. 3 80335 München, Germany Email: dgruber@univa.com

6 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.

7 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.

8 Full Copyright Notice

Copyright © Open Grid Forum (2014). Some Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included as references to the derived portions on all such copies and derivative works. The published OGF document from which such works are derived, however, may not be modified in any way, such as by removing the copyright notice or references to the OGF or other organizations, except as needed for the purpose of developing new or updated OGF documents in conformance with the procedures defined in the OGF Document Process, or as required to translate it into languages other than English. OGF, with the approval of its board, may remove this restriction for inclusion of OGF document content for the purpose of producing standards in cooperation with other international standards bodies.

The limited permissions granted above are perpetual and will not be revoked by the OGF or its successors or assignees.

9 References