

Grid-Enabled Desktop Environments

The GRENADE project

Stephen Pickles

UK e-Science All Hands Meeting
Nottingham, 2-4 September, 2003.



THE UNIVERSITY
of MANCHESTER

The web of the future?

- Where is the Grid's killer app coming from?
- How can the Grid deliver its promised ubiquity within the confines of the community that conceived it?
- Can we enlist the skills and imagination of a much wider community to accelerate the development of the Grid?
- The DataGrid has been called "Napster for scientists". Is it good enough for particle physicists but not friends and family?

Get the geeks engaged!

IMHO: User Interfaces to the Grid

- Portals are great, but...
 - They break the symmetry between local and remote resources
 - Why can't I use the same user interface to browse remote files as local ones?
 - Can a third party extend the functionality of the portal?
 - Lesson: make middle tier services accessible!
 - A single portal provides uniformity of presentation, but a user must use many portals...
 - Where should persistent user context reside?
 - Sometimes (eg. interactive visualization) a thin client won't do
 - Or will it? Can TightVNC help deliver this through a web portal?
- Command line interfaces are powerful, but arcane
- Desktop metaphor has been neglected

The Vision

- Robust, interoperable Grid toolkits ship with every PC
 - We were promised ubiquity
 - **STOP PRESS: SUSE Linux now ships with GT2**
- Grid functionality on every desktop
 - accessibility, visibility, public awareness
- Ordinary people use Grids, not just e-Scientists
- Desktops consume *and export* Grid services
- Thriving open source Grid projects
 - games, file sharing, code sharing, "friends and family grids"...

How do we get there from here?

The Plan

GRENADÉ is exploring the possibilities offered by tight integration of Grid functionality into the user's desktop.

Two phase strategy:

1. Develop a prototype Grid Enabled Desktop (demonstrator) to pump-prime an on-going open source project,
2. Support the open source project and encourage emulation on other platforms.

Target KDE in the first instance

- Shipped with most Linux distributions
- User base (Linux, Irix, AIX, Solaris,...) and portability
- Component-based architecture with XML descriptions
- Konqueror browser designed to integrate plug-in components

But will choose more portable Qt classes instead of KDE classes wherever practical

Curious observations

Early versions of Globus had

- poor separation between client and server bundles
- APIs limited to C and command line.

(These are "fixed" by better packaging, COG-kits and portals.)

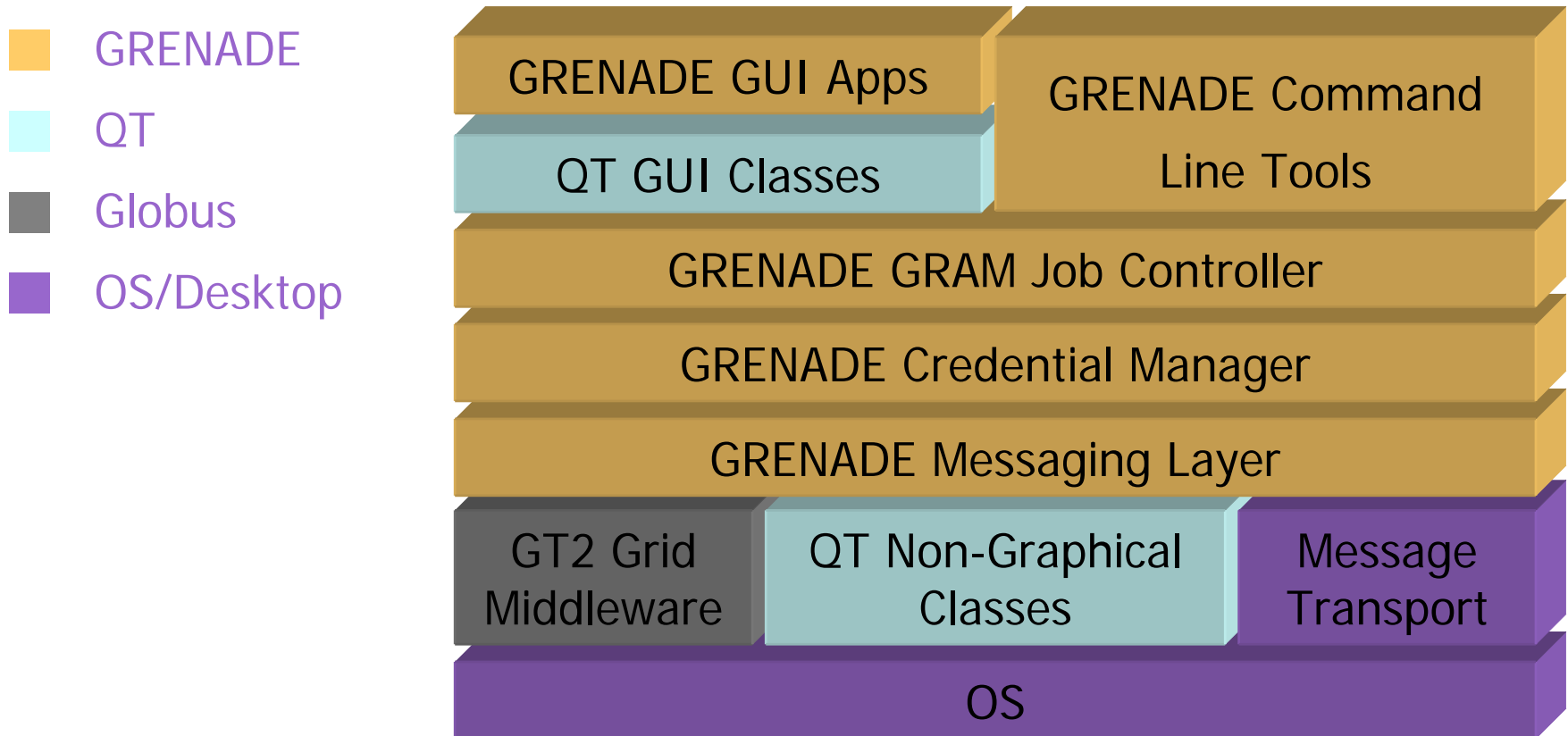
GSI suggests having a single base for Grid activities where your private key is stored. The obvious place is on your own PC or laptop. So why isn't this the norm?

Curiously, none of these presents an obstacle to GRENADE.

By demanding a complete Globus installation, GRENADE can

- use Globus API's and drill down into host's native API layers
- provide – and facilitate management of – Globus services.

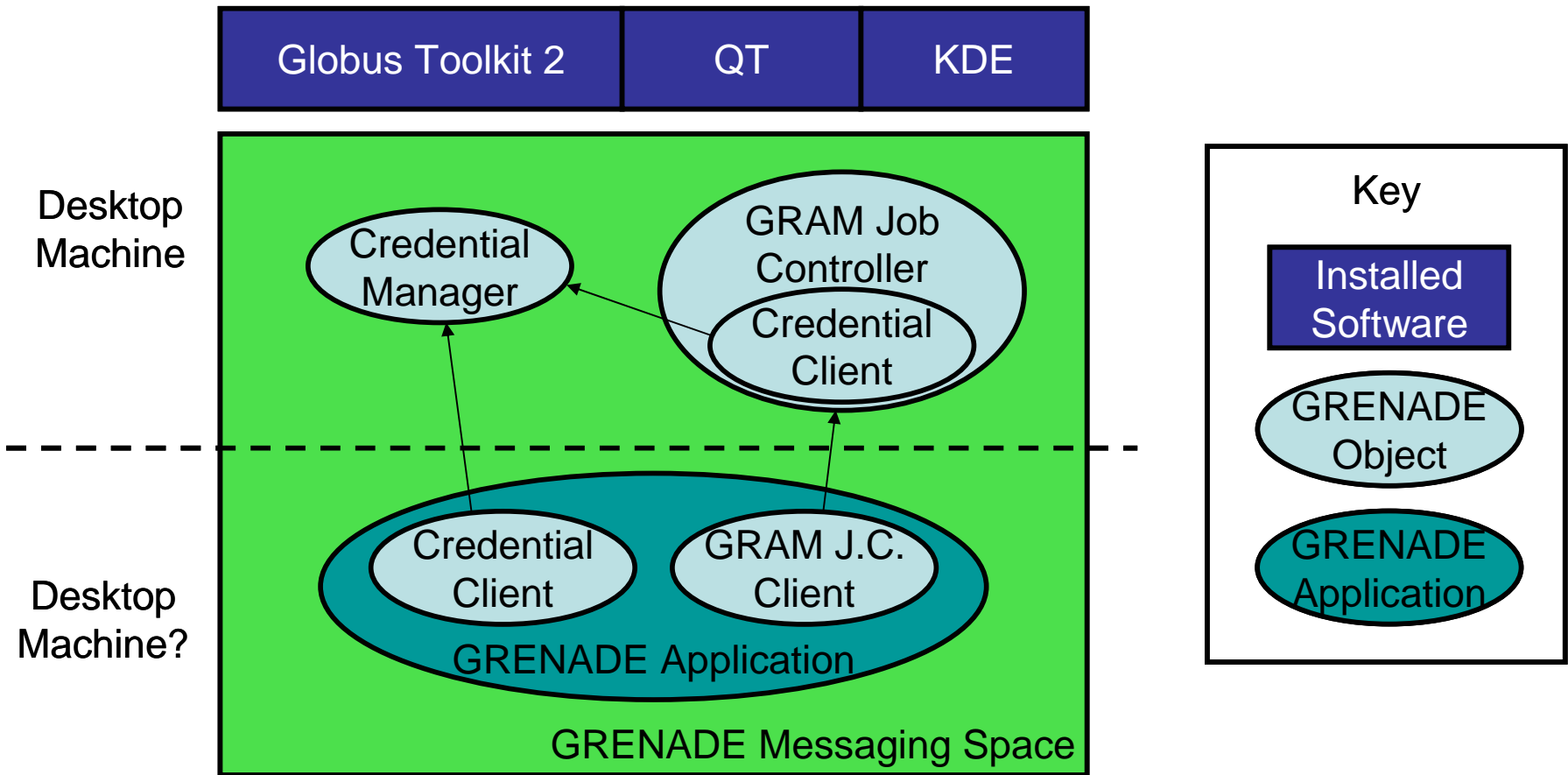
GRENADe Software Stack



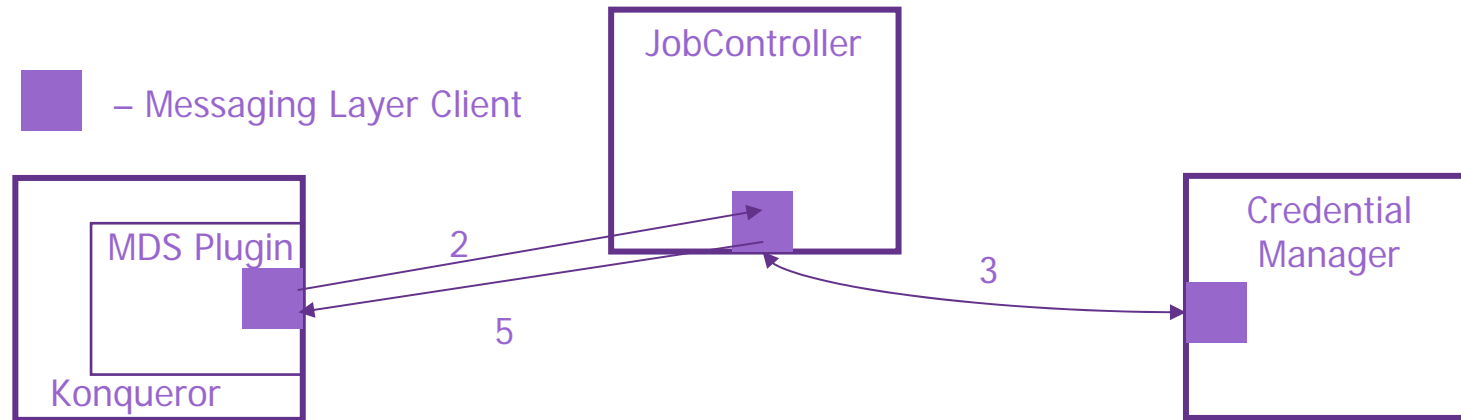
GRENADe Stack

- **Messaging layer**
 - Basis of the architecture is a simple messaging layer for inter-application communication
 - originally just intra-desktop, but we've seen the light!
 - Sends serialized objects over QDataStream (essentially a stream of bytes)
 - Can be implemented over different transports, e.g. DCOP, SSL.
- **Credential Manager**
 - Looks after the user's X509 certificates and keys
 - Can generate GSI proxies
 - Each credential has a simple name or alias identifying it, e.g. "ukes" or "globus"
- **GRAM Job Controller**
 - Remembers and looks after jobs launched via GRAM
 - Each job has a simple name or alias identifying it, e.g. "rg-sim"

GRENADe Architecture



Example Usage



1. A user drops an RSL description of a GRAM Job onto a Globus JobManager in the MDS Browser (Konqueror plugin)
2. The plugin sends a "start new job" message to the JobController Application, containing the RSL string and JobManager URL
3. The Job Controller selects the credential to use for that JobManager, from the list obtained from the Credential Manager
4. Job Controller contacts Globus JobManager using GRAM to start the job.
5. The Job Controller sends a status message back to the client, plus a job id

GRENADÉ Command Line

- Adds value to underlying Grid middleware
 - Names/aliases provide simple reference mechanism for both credentials and jobs
 - Can list, add to, remove credentials from the manager
 - Can list known GRAM jobs, find out how they were started, etc.
 - Can refer to GRAM jobs without using horrible URL
- Insulate the user from hard-to-use things, like:
 - Proxy creation for non-standard location
 - Inconsistent GT2 command line interface

Deliverables

- Reference Implementation – GRENADE 1.0 (Q4 2003?)
 - Single-sign on
 - Job definition, submission and monitoring tools
 - MDS browser, pluggable into both Konqueror and job controller
 - Remote file-system browser (teach Konqueror to speak GridFTP)
 - Validated on Linux and Irix
 - Documentation
- White paper on Grid Enabled Desktop Environments
- Support open source project
 - 1 day/week for 12 months
 - Web site, mailing list and technical support
 - Bug fixes, enhancements and new functionality as resources permit
- Release of GRENADE 1.1
- Final report

- GRENADE launched with workshop in December
- Behind schedule
- More comprehensive design than originally planned
- Command line interface is new
- Design is future-proofed against requirements for
 - Drag-and-drop job submission
 - Support for multiple credentials
- Implementation in progress

Possible extensions

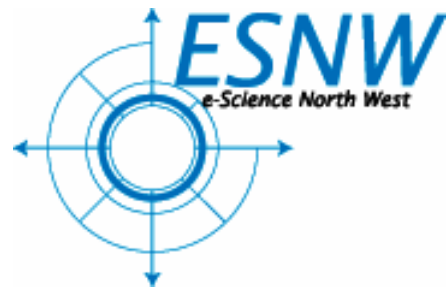
- xterm/gsi-ssh
- Drag-and-drop job submission
- Support for multiple credentials (singleton pattern in GRENADE 1.0)
- GridFTP GUI, supporting 3rd party file transfers, remote-remote
- Resource broker integrated with job submission GUI
- GUI's for user management
- File system synchronisation
- Expose GRENADE services as OGSi services
- GT3 integration
- Port to Windows – messaging layer easy, Explorer integration harder, Globus could be the show-stopper.
- Basis for P2P desktop Grids
- Your suggestion here!

GRENADe: people and sponsors



Manchester Computing, e-Science Team:

- Stephen Pickles, Principal Investigator
- Jon MacLaren, Software Architect



Advanced Interfaces Group,
Dept of Computer Science:

- Steve Pettifer, Co-Investigator
- James Marsh, User Interfaces



THE UNIVERSITY
of MANCHESTER



SGI:

- Andrew Grant, Principal Contact
- Martyn Foster, Technical Contact

SVE @ Manchester Computing

World Leading Supercomputing
Service, Support and Research

*Bringing Science and
Supercomputers Together*

www.man.ac.uk/sve
sve@man.ac.uk



THE UNIVERSITY
of MANCHESTER