Jini and MindStorms

Jan Newmarch
jan@ise.canberra.edu.au
University of Canberra
DSTC (Distributed Systems Technology CRC)

Jini and MindStorms

- Jini brings ‘‘network plug and play’’ to hardware devices and software services
- Jini is in pure Java
- It makes hardware and software available as Java networked services
- ‘‘Legacy’’ systems can be wrapped in a Java proxy
- Jini has competitors including CORBA Trader, DCOM, HP’s Chai, Enterprise Java Beans, JNDI, etc

 © Jan Newmarch

Jini Federation

- A Jini federation or ‘‘djinn’’ consists of a number of clients, services and service locators
- Typically connected by TCP/IP
- Code is moved around from service to locator to client using Java 1.2 extensions to RMI

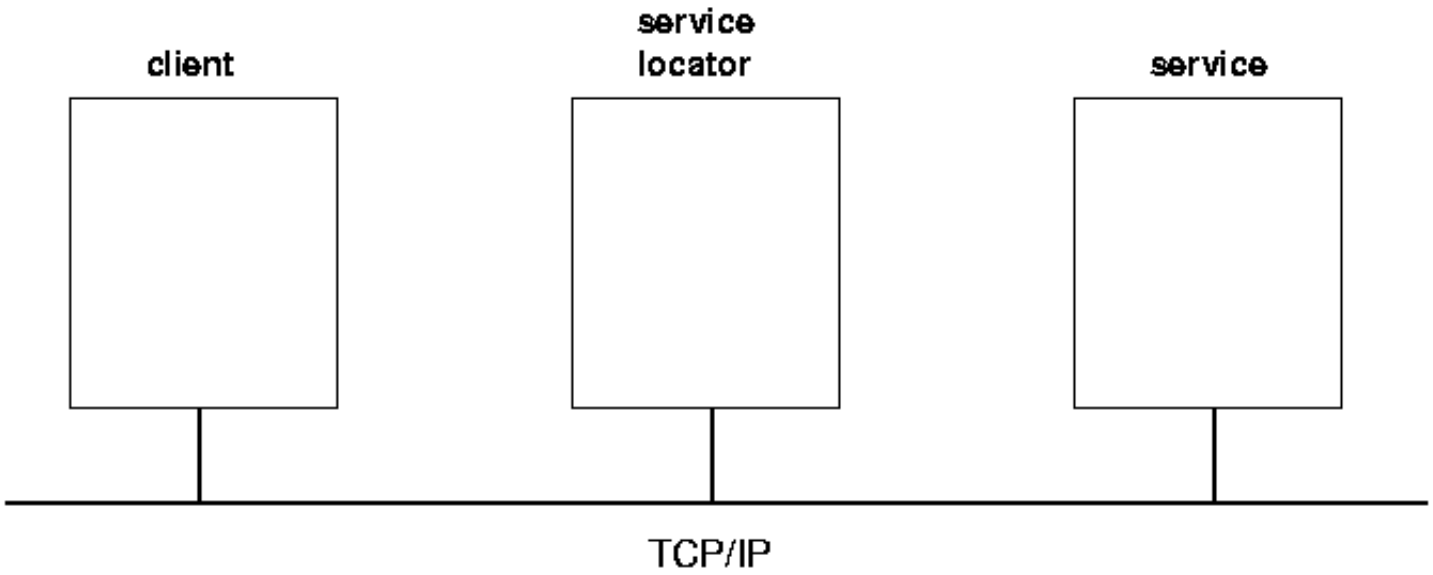


Figure 1: Components of a Jini federation

 © Jan Newmarch

Service Registration

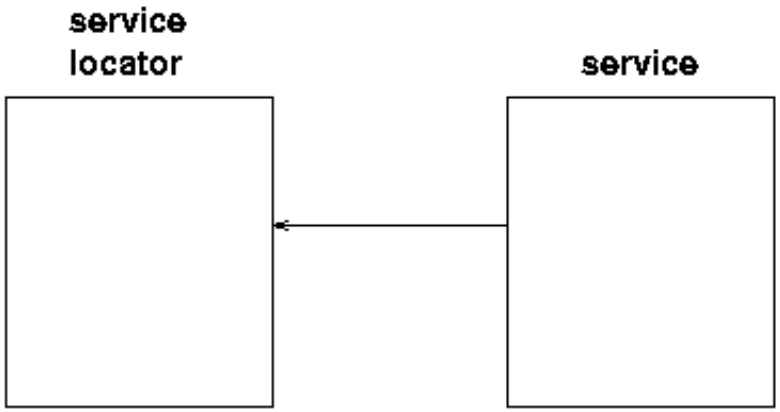


Figure 2: Querying for a service locator

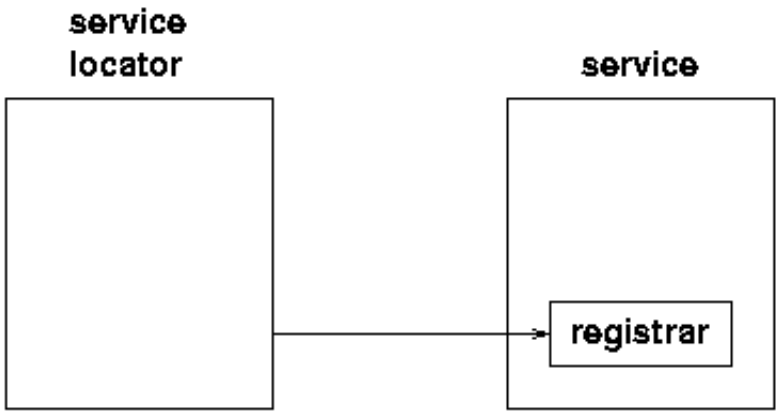


Figure 3: Registrar returned

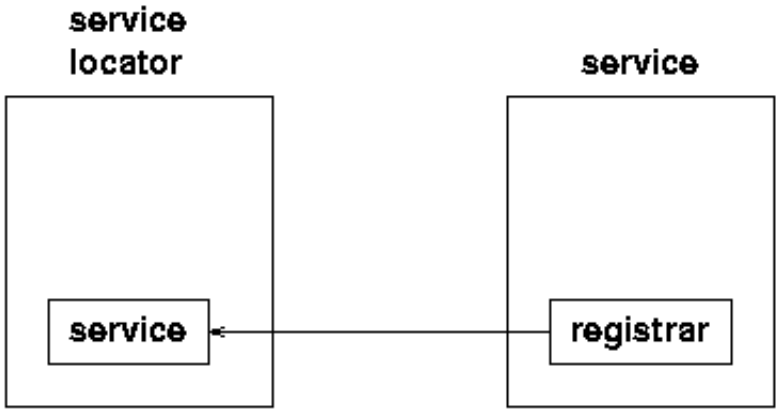


Figure 4: Service uploaded

 © Jan Newmarch

Client Lookup

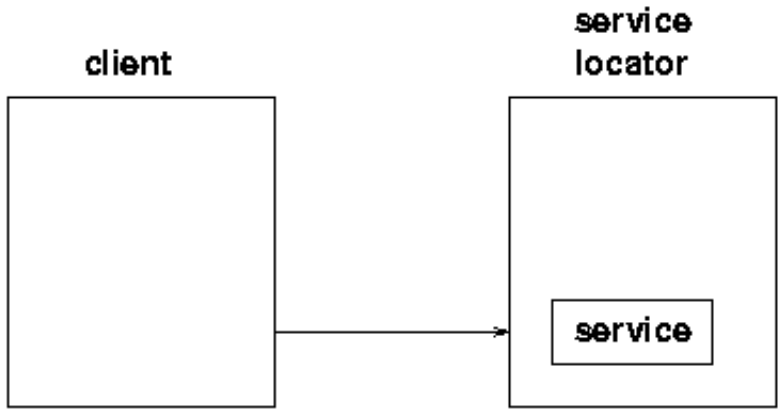


Figure 5: Querying for a service locator

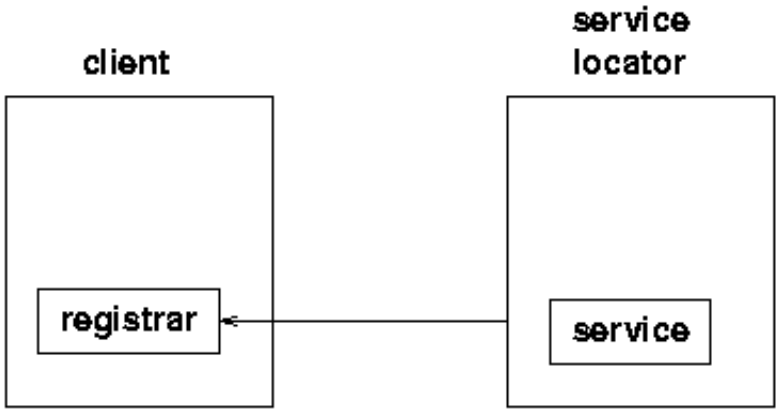


Figure 6: Registrar returned

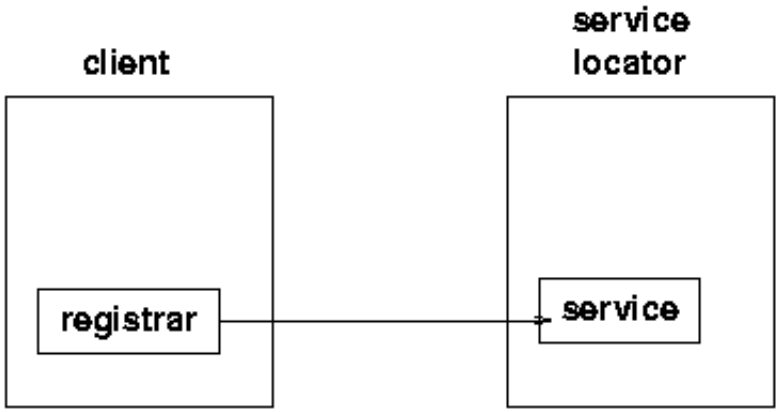


Figure 7: Asking for a service

 © Jan Newmarch

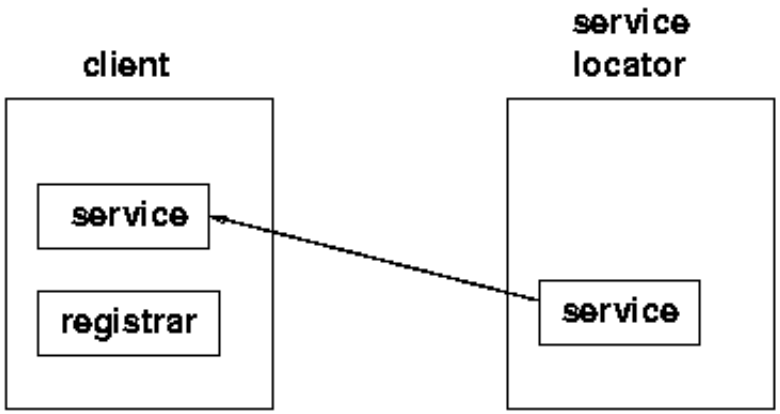


Figure 8: Service returned

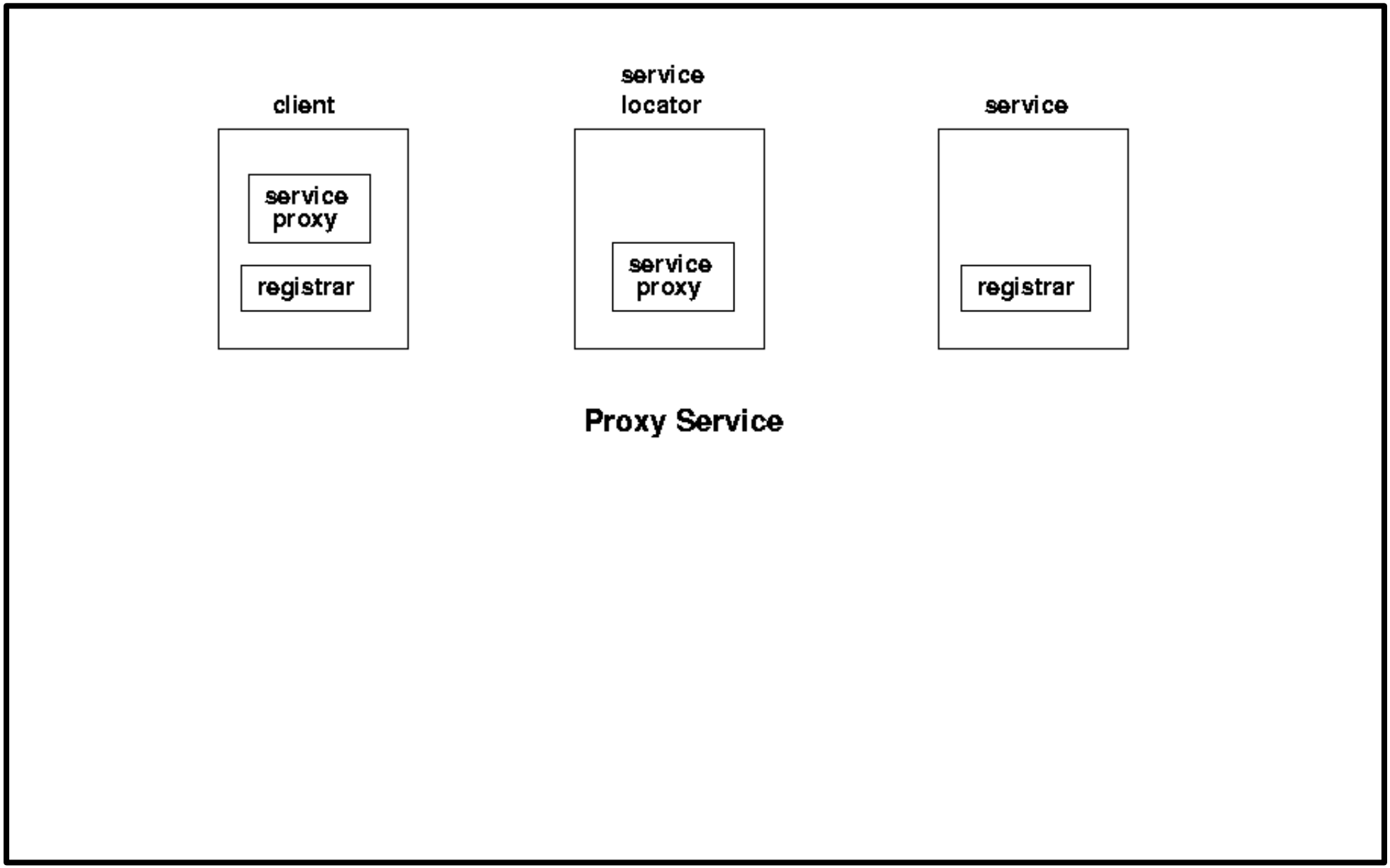
Support Services

- Jini objects are moved in ‘‘marshalled’’ form
- Class files may need to follow these for proper activation
- The class files may need to be available from HTTP servers
- RMI Naming can be used to locate remote objects
- RMI Naming may need an rmiregistry or rmid running
- Security permissions will need to be set using JDK 1.2 security model

 © Jan Newmarch

Proxies

- A toaster will be controlled by some service software
- The service software will be moved across the network to the client
- The mobile code must be a proxy for the real service code
- The proxy may be explicitly written or be an RMI proxy stub
- Remote calls from the proxy may be RMI Remote calls, or by some other mechanism such as private TCP connection



 © Jan Newmarch

Attribute Registration

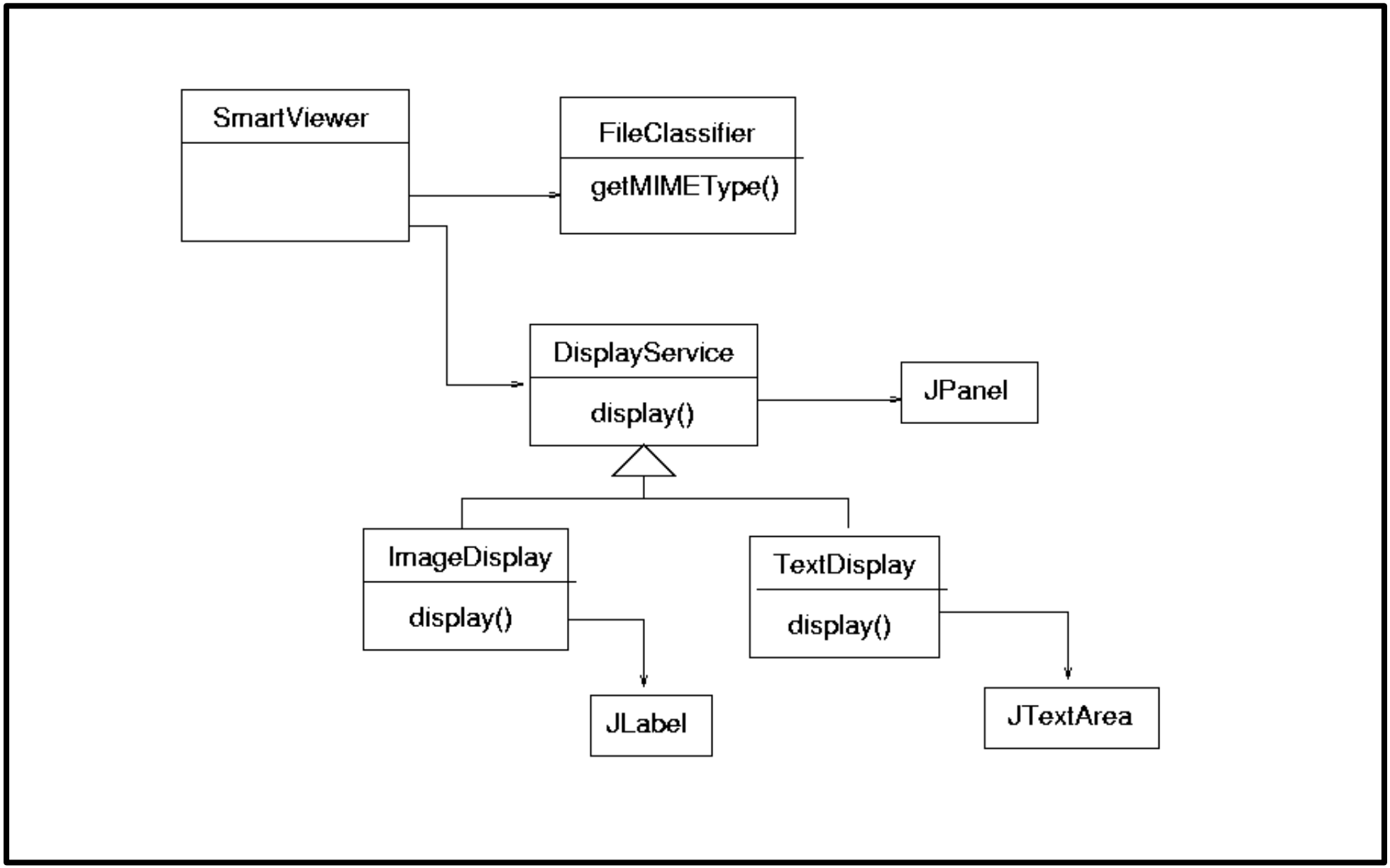
- A service is uploaded to a locator as an instance of a class
- Additional information such as address may also be uploaded as instances of Entry classes
- Service registration is done using a leasing system

Service Location

- A client asks for a service as an instance of a class, typically of an interface
- A client may ask for additional Entry attributes
- Matching on fields is *exact*
- You can ask for a printer with a certain speed, but not for one *greater than* a certain speed
- Additional services may be used if complex boolean expressions are needed

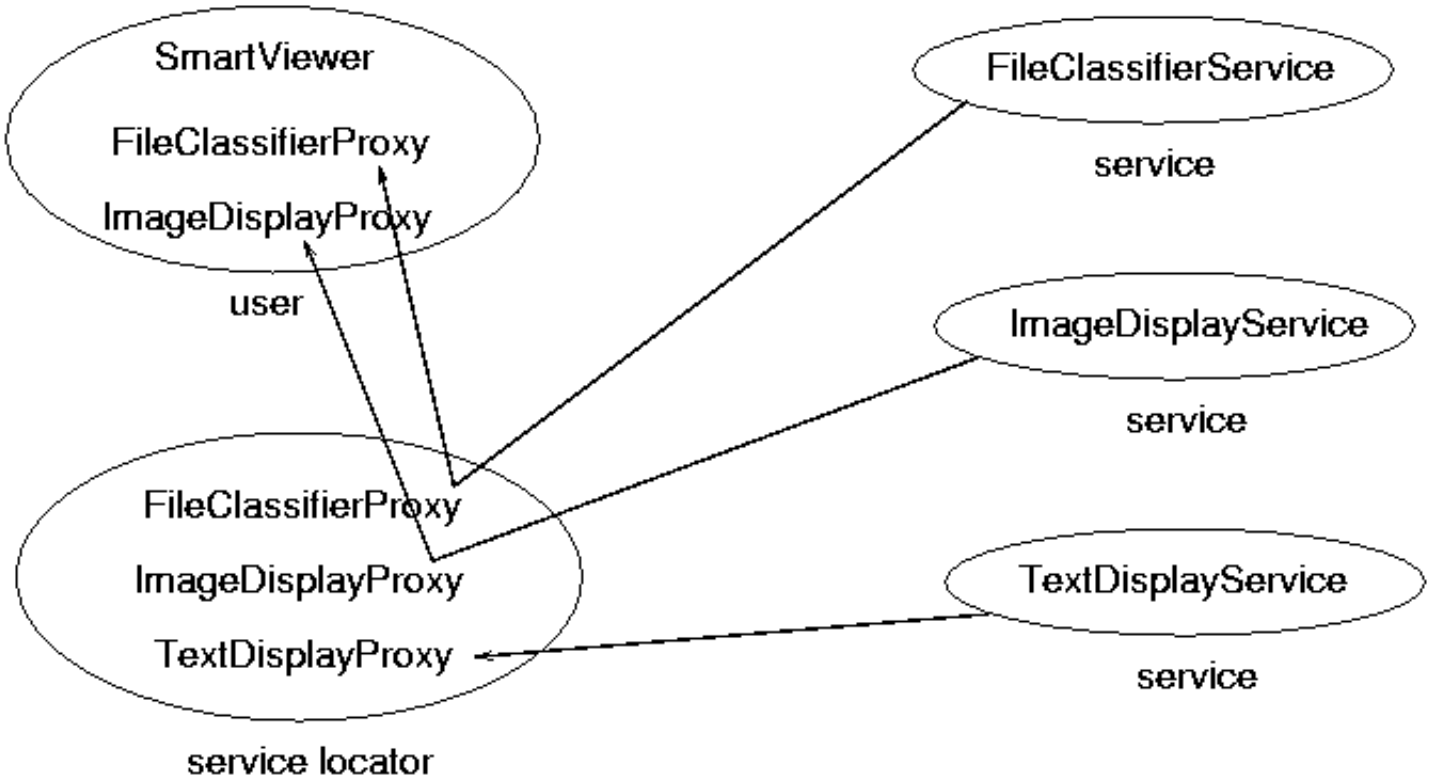
 © Jan Newmarch

Non-distributed Application



 © Jan Newmarch

Distributed Version



Problems and Issues

- Objects moved around must be serializable
- Some common objects are not serializable e.g. Swing Toolbar, Swing JTextArea
- Objects with DLLs are not serializable e.g. MPEG players
- JDK 1.2 memory requirements are enormous compared to JDK 1.1
- Jini v1.0 has some bugs
- Limited documentation

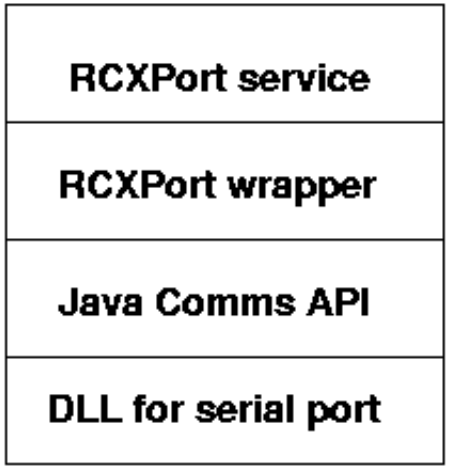
 © Jan Newmarch

Lego MindStorms

MindStorms is a ‘‘Robotics Invention System’’

- A 16-bit computer called the RCX (a 16MHz Hitachi H8/3297), with 32k RAM
- The RCX can control 3 motors and read from 3 sensors
- Additional Lego parts - gears, wheels, sensors, motors
- Software can be downloaded by an infra-red link on a serial port
- Programs can be downloaded and run, or ‘‘immediate’’ commands can be run
- Normally programmed from a visual programming environment or from VB
- On a PC, interface to RCX is managed by an OCX
- Low-level machine-code is sent across the IR link

MindStorms as a Jini Service

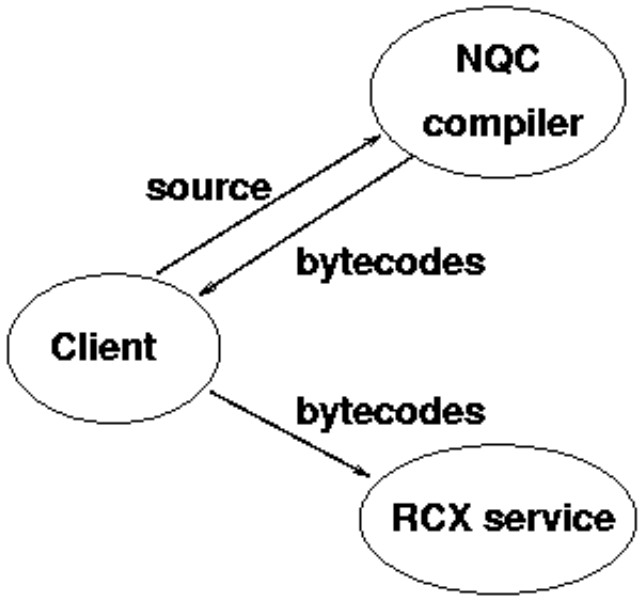


RCX Jini Service

 © Jan Newmarch

Future MindStorms Work

- Encapsulate client-side in a convenience class
- Make the ‘‘Not Quite C’’ RCX compiler available as a Jini service



- Make the RCX ‘‘network aware’’, by using e.g. an alternative O/S such as LegOS

 © Jan Newmarch

Future Work

- Build bridges to other network services (CORBA Trader, JNDI, etc)
- Interface to other network protocols and implementations (BlueTooth, HomeRF, etc)
- Design standard interfaces to service types
- Build Jini proxies for hardware and software services
- Build embedded Jini engines (in Java or other languages) for hardware devices
- How does ‘‘service architecture’’ differ from client-server, CORBA, etc?

URLs

- Jini: http://www.jini.org
- Jini FAQ: http://www.artima.com/jini/faq.html
- Lego MindStorms: http://www.legomindstorms.com
- Unofficial Lego MindStorms: http://www.crynwr.com/lego-robotics/
- Jini tutorial: http://pandonia.canberra.edu.au/java/jini/tutorial/Jini.xml
- Motorola Piano: http://www.mot.com/GSS/SSTG/piano/

 © Jan Newmarch

Jan Newmarch (http://pandonia.canberra.edu.au)

jan@ise.canberra.edu
Last modified: Fri Jul 16 12:02:32 EST 1999
Copyright ©Jan Newmarch