

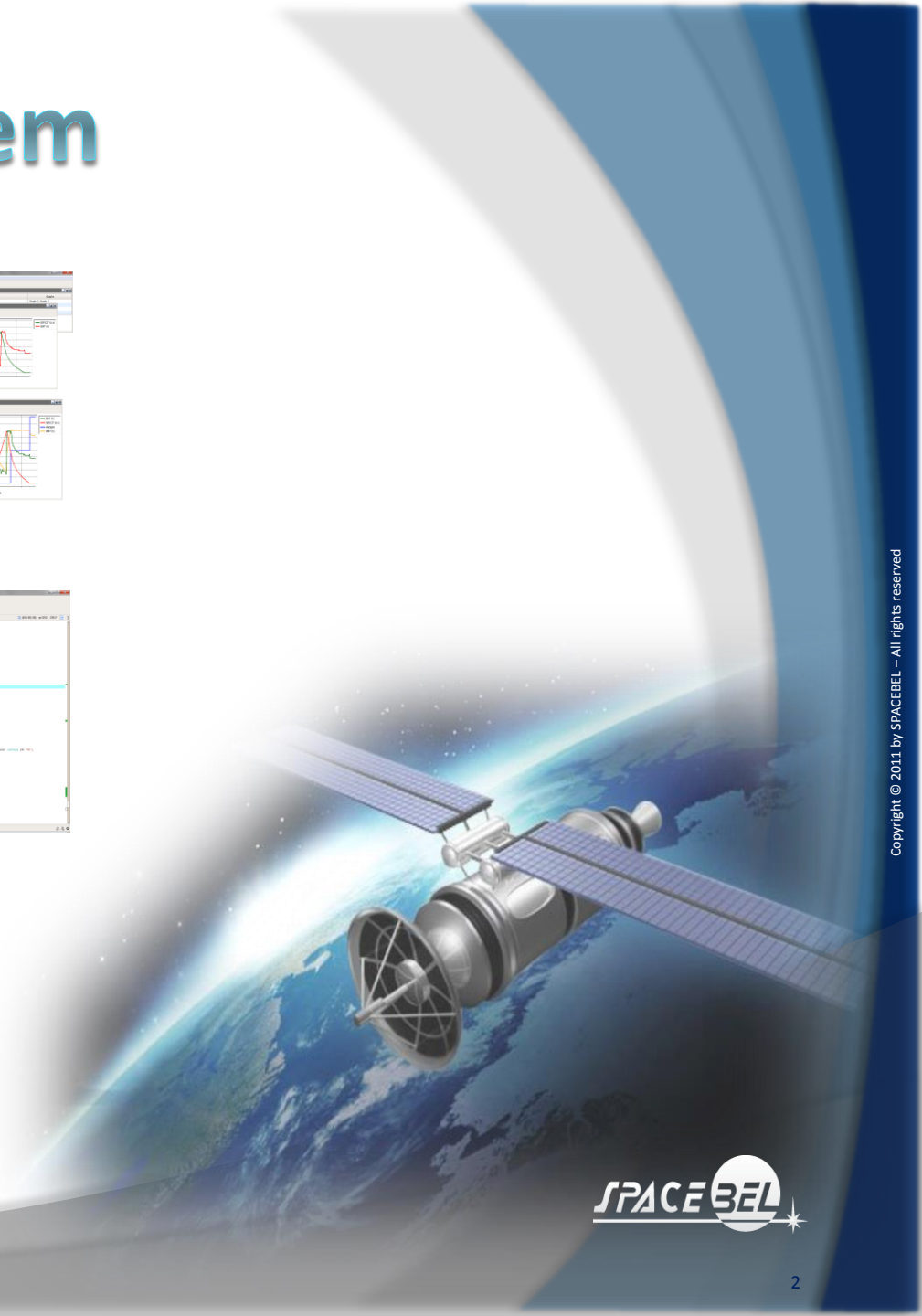
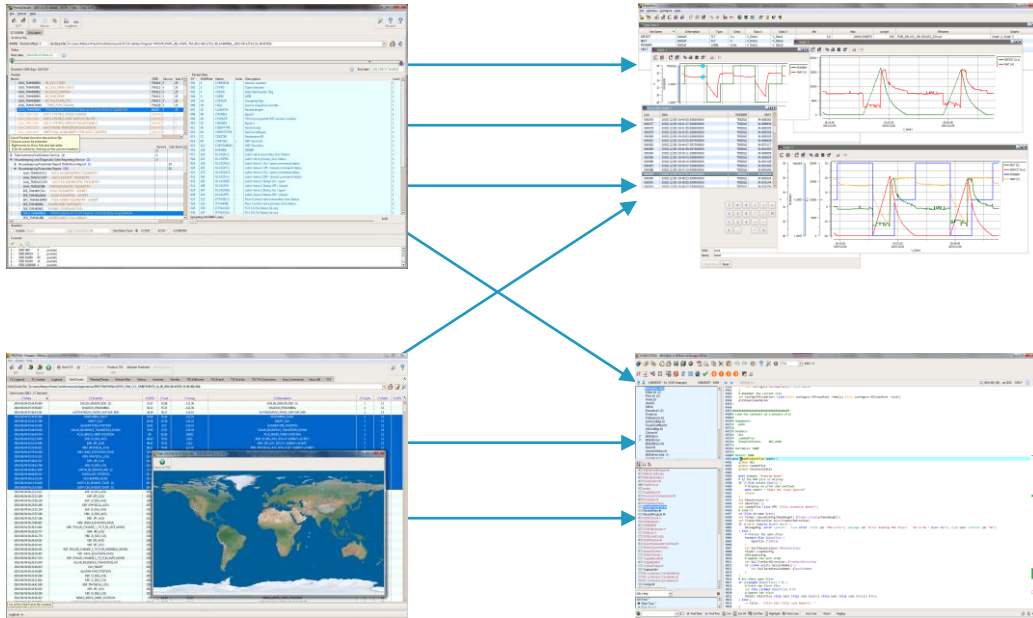
SESSIONS

A simple way to handle Single/multiple sessions

Martyn SMITH



The Problem



The First Solution

Single Socket server first come first served

- Needed a program restart to change state
- Port number needs to be defined for each user/server
- Can not have multiple identifiable instances



The Ideal Solution

- Multiplatform Multiuser on Linux
- No Central Server with automatic discovery
- Multiple identifiable sessions
- Dynamic changing of session IDs
- Stable, if a program crashes



Discovery

How can two programs find each other?

~~Most~~ ~~Simplest~~ using

```
wininfo name APP:portid      wininfo interps
```

DDE using

```
dde servername APP:portid  dde services TclEval {}
```



Solution

- Every application starts a server
- All publish their portid using winfo/dde
- Broadcast always fetches the port list
- If there is a default name, ask if used
 - If not USED set as current name
 - If USED send argv and exit
- If the default changes tell everyone
- If the user changes name, ask if used
 - If not USED set as current name
 - If USED refuse change

Command line option ‘–session NAME’

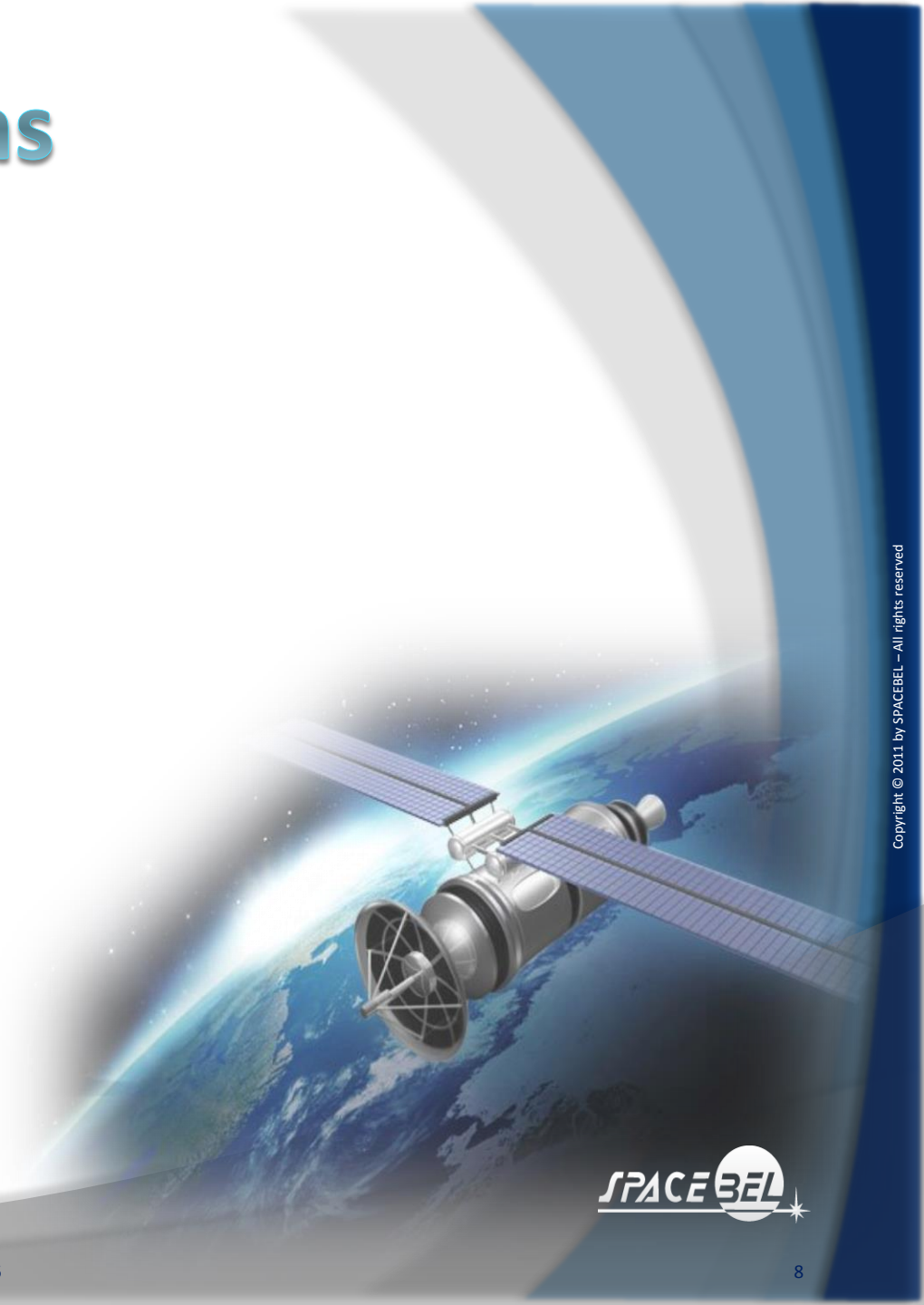


Protocol

All messages have the header
ServerID:USER:PID@HOSTNAME

Broadcast	Received
<i>EXIST</i>	If the name is ours reply <i>USED</i>
<i>DEFAULT</i>	Change our default if our INI
<i>SENDNAME</i>	Return our <i>NAME</i>
<i>OPENFILE</i>	If the name is ours open the file
...	
<i>DATA</i>	Send program data between sessions

Questions



Copyright © 2011 by SPACEBEL – All rights reserved

