

Whither RPSL?



i n e x
internet neutral exchange

Nick Hilliard
CTO - INEX
nick@inex.ie



i n e x
i n t e r n e t n e u t r a l e x c h a n g e

Let's rewind to 2009



i n e x
i n t e r n e t n e u t r a l e x c h a n g e

Lisbon Meeting



IRRToolSet



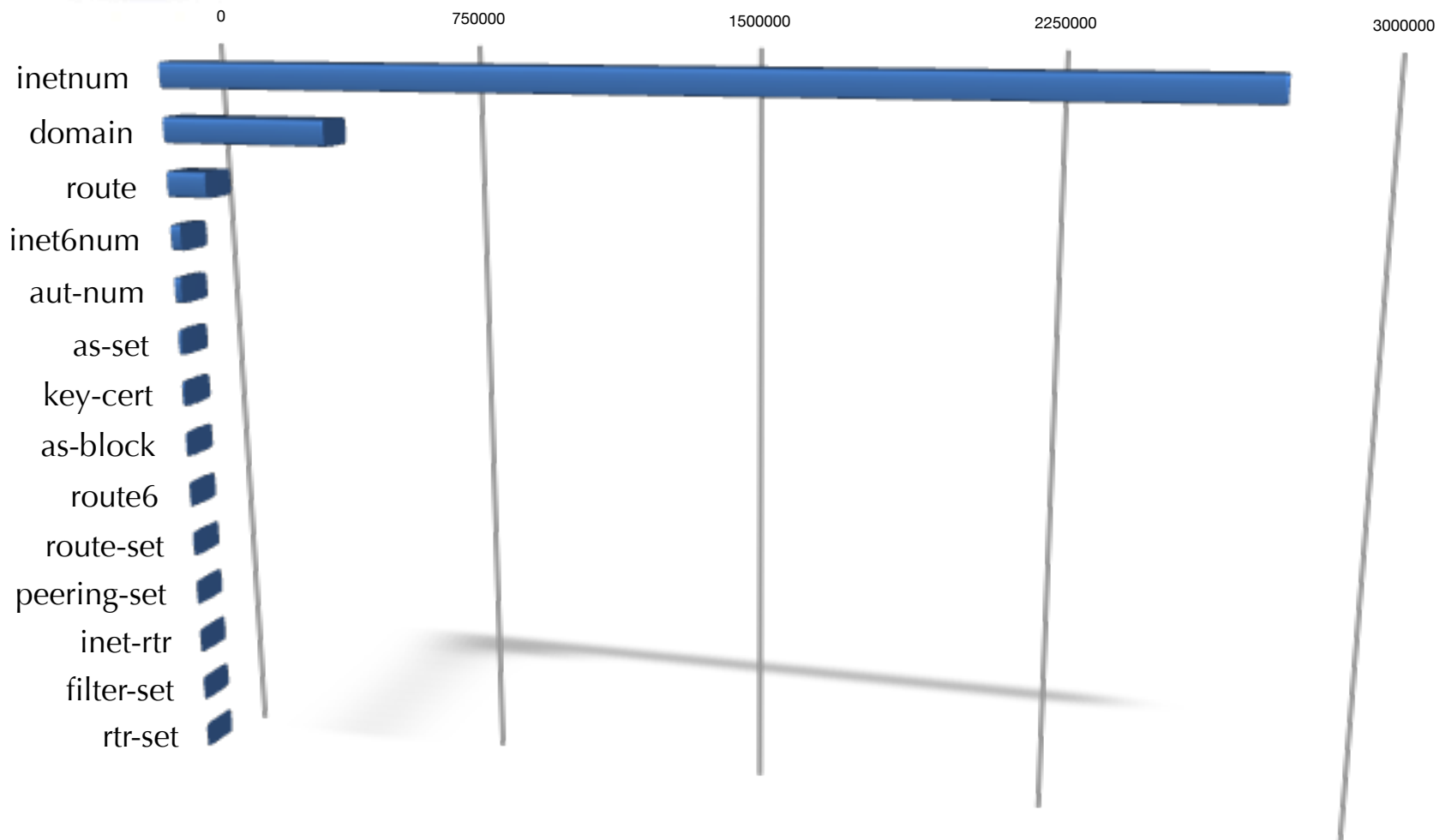
i n t e r n e t n e u t r a l e x c h a n g e

...but we also had
some data



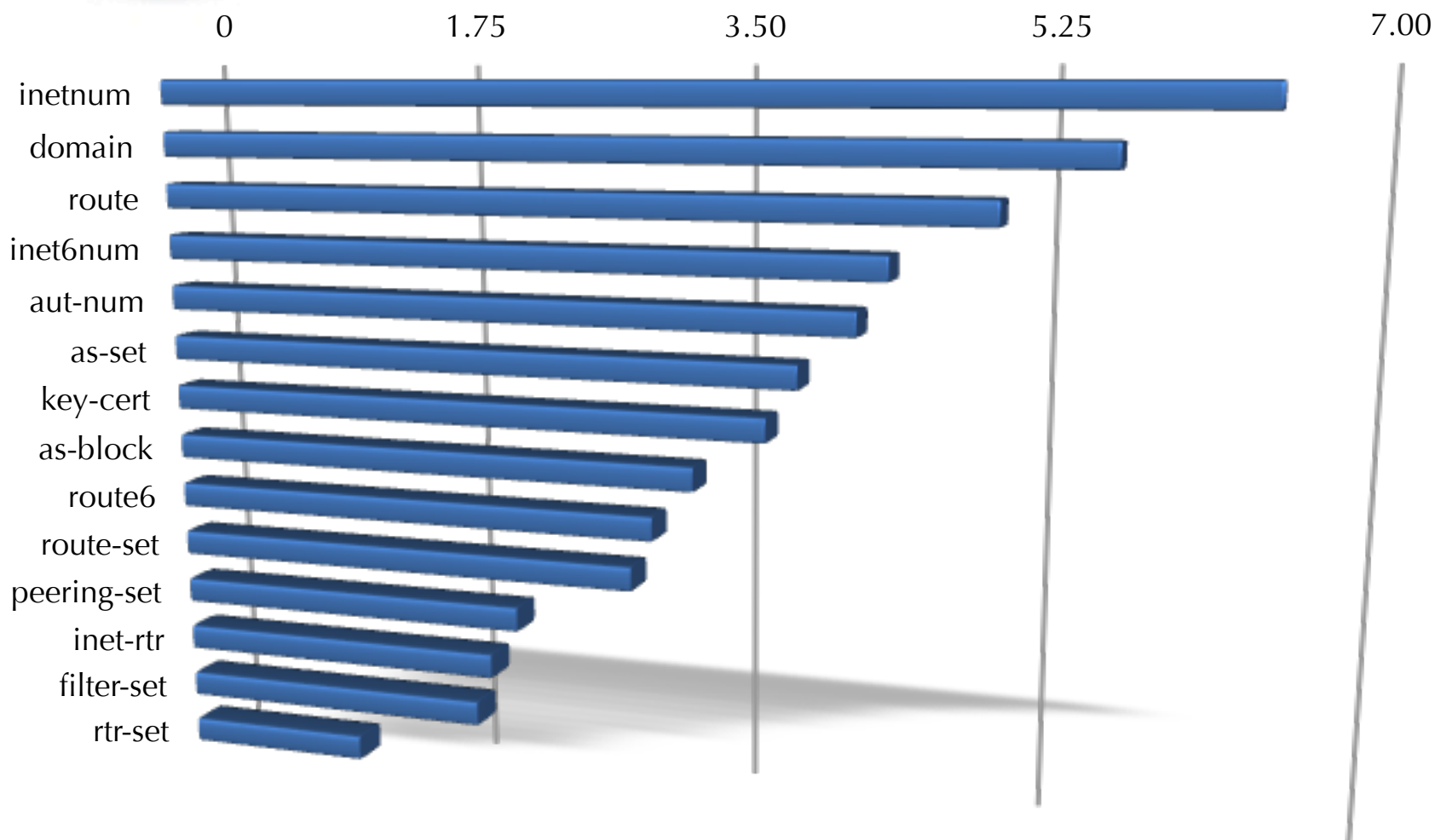
i n e x

i n t e r n e t n e u t r a l e x c h a n g e





i n e x
i n t e r n e t n e u t r a l e x c h a n g e





i n e x
i n t e r n e t n e u t r a l e x c h a n g e

IRRDBs are a success

...but not in the way that was originally envisaged



User Experience of RPSL

i n e x
i n t e r n e t n e u t r a l e x c h a n g e

- Most people use tiny pieces of it
- Most people don't understand its capabilities
- Most people agree that RPSL is necessary
- Only IRRToolSet has reasonably complete RPSL implementation
- No-one understands IRRToolSet code
 - Almost impossible to augment/fix/debug the code
- The approach is wrong
 - start with script bindings and link to Template tool
- Other toolkit approaches are much better
 - rpsltools: perl code + Template::Toolkit



RPSL Deficiencies

- Specification is too complicated to parse easily
- But too limited to allow full router configuration
- Doesn't support modern address families
- Doesn't cope with interior routing
- IPv6 support added as afterthought
 - implementation is slightly messy



Changing the RPSL Specification

i n e x
i n t e r n e t n e u t r a l e x c h a n g e

- Simplification of language
 - would make it much easier to write tools
 - would make it much easier for end-users
 - would probably break some policy specs
- Augmenting the RPSL language
 - would make it even more difficult to write tools
 - doesn't solve the usability problem
 - would probably be backwards compatible



Ways Forward

i n e x
i n t e r n e t n e u t r a l e x c h a n g e

- Re-examine RPSL from ground up
 - i.e. trawl through RIPE Database
 - ... realising that some users have private databases which contain all sort of weird things
- Extract some meaningful information on what people are using RPSL for
- Examine whether RPSL could be simplified, and if so, how much work would it take
- If not possible, then consider research into a new language to take the place of RPSL



End Goal

i n t e r n e t n e u t r a l e x c h a n g e

End up with language that people can use

End up with language that people can write code for

End up with a 95% solution rather than a 100% solution



i n e x
i n t e r n e t n e u t r a l e x c h a n g e

Discuss