

keynote

9.30 - 10.50

Date 6, 9

1. Open source is getting more prevalent
2. We need more investment
3. Public money public code
4. We need ways to support users with our stuff
5. Big Tech keeps raising prices
6. Open source costs don't raise so fast
7. Some ministers like Plasm (Germany)
8. Changing minds $\Rightarrow$ not outsmart
9. Focus right direction not doctrine
10. White with right people
11. Small steps
12. Most arguments against moving are non-technical
13. Contacting: website $\Rightarrow$ mail, join events, talk to them
14. it's not that complicated
15. Find right people
16. Ask for correct people to talk with
17. Schools are best entrypoint for change
- 18.
- 19.
- 20.
- 21.
- 22.
- 23.
- 24.
- 25.
- 26.
- 27.

Lessons learned 1130-1205

Date 6, 9

1. People still talk of KDE 4.
2. split things up if you can
3. Policy not working extreme opposite isn't answer
4. Youtubers are ambassadors
5. Don't add "wouldn't it be cool if..." feats
6. Some workarounds are inevitable
7. File Qt bugs
8. No more gimmicks!
9. Wouldn't it be helpful if.. ← good



Banana OS

13:35 - 14:00

Date 6 / 9

1. etc - factory

ERFS + BTRFS

2. bclean /etc

3. bc factory reset

4.

5. KDE linux alpha! today!

6.

7. Delta download

8. splits DL to chunks

9.

10. Local link

11. share downloads between computers
12. in same network

13.

14. Panamapack (Flatpak exploration)

15. Remove sandbox?

16.

17. Obstsalat

18. many pieces, kiss, backups, etc...

19.

20. product by us for us

21.

22. Harald is giving out bananas

23.

24. Do what the app wants you to do

25.

26. Talk was bananas...

27.

Qt different langs 14:15-14.55 Date 6.9

1.

2. Approach: Bindings

3.

4. C++ is flexible, not easy for bindings

5.

6. Python is used in many places

7.

8. Pyside ~ writing Qt app from scratch
9. → its a lot for newbies

10.

11. Qt bridges → avoid above complexity?

12.

13. Bindings Bridge

14.



15.

16.

17. Internal process → Qt's Object process

18. Implementation details → langs use different
19. details

20. Dev environment shouldn't change

21.

22. Target users: new devs

23. Bridges may not allow everything like C++

24.

25. People could create their own Bridges

26. some features will not be considered

27. Need more? C++ lol (or bindings)

28.

28. No widgets

29. Not every Qt feature

30.

30. Open source? yes

31.

32. Release? Maybe first release at november

33.

1.

2. Union "unionizes" widgets, quick, plasma

3.

4. kirigami integration for units

5. overrides to allow theming kirigami

6. still some trouble with some controls

7.

8. SVG is not future proof

9.

10. CSS: no logic

11. easy to change

12. extensible

13.

14. Servo?

15. - we use their CSS parser

16. to make our own parser

17.

18. CXX-Rust-CSSparser: wrapper to give C++
19. an abstract resemblance

20.

21. Selectors: type: "button"

22. button { }

23. properties: colors etc.

24.

25. Functions: need to be implemented

26. CSS is hard work but future proof

27. First Release ASAP

28. Plasma Next made with Union

29. TODO: at widgets, decorations

30.

1. Why c++? Elders decided, Qt
2. Why not? complex! Footguns!
3. Rust! Ergonomic, memory safe
4. Mixing c++ and Rust
5. — cxx crate bidirectional
6. — cxx-qt Doesn't wrap everything, for Qt
7. — Corrosion for cmake
8. Rust in use
9. — AngelFish and PIM adblock
10. — PIM HTML parsing
11. — Union CSS parsing
- 12.
13. cxx-kde-frameworks - Rust Frameworks APIs
14. — Builds on top of cxx-qt
15. — KAboutData
16. — Translation
17. — Crash handling
18. — KCMS
19. — Soon: KIO workers!
20. ↳ Move previewjob to Rust someday?
21. Rust is not perfect! Hard to approach, rigid
22. Not complete Qt support
23. Not ideal for app glue
24. Python: Approachable, Good Qt Interap
25. Pyside
26. KDE frameworks python bindings: covers a lot!
27. Rust and Python are good fit but won't replace C++
28. Python distribution is bad
29. No "Rewrite in Rust"
30. Rust is good at low-level, security places
31. Leverage crates
32. Risks of technical and social fragmentation
- 33.

1. Open source but open to who
2. Burn out is seen as token of prestige
3. Stakeholders: anything affected
4. Every MR = whose voice gets merged
- 5.
6. What world is your project making possible?
- 7.
- 8.
- 9.
- 10.
- 11.
- 12.
- 13.
- 14.
- 15.
- 16.
- 17.
- 18.
- 19.
- 20.
- 21.
- 22.
- 23.
- 24.
- 25.
- 26.
- 27.

Maintainers

13.30 - 14.00

Date 7, 9

1. Getting contributors:
2. Do cool stuff
3. Tell people about it
4. Let them join
5. Maintainer leaving:
6. Project dies
7. No authority
8. Power struggle → infighting, forks
9. Look for replacement while still an project
10. Good maintainer: communicate, custodian?
11. Future maintainer:
12. - knows a topic
13. - feedback friendly
14. - partakes in other tasks like bug triage
15. start with reviews (New maintainers)
16. Explain rationales
17. Empower, trust them make decisions
18. - can still be influence
19. Provide roles like sub/co maintainer
20. Encourage them to join meetings
- 21.
22. Learn to let go and give space
- 23.
24. Watch out for buses! ☹
- 25.
- 26.
- 27.

Qml data 6.10 17.18-14.50

7 / 9

1. model delegate view can have problems with mismatched delegates

2. Can be fixed with required properties

1. Clean errors

2. easy reuse

3. Does not allow update updating model (before qt 6.10)

4. after 6.10, use `delegateModelAccess` on the view

5. `delegateModelAccess: DelegateModel.ReadWrite`

(use model.value
but with this
we get the type)

6. Available for all views shipped with qt

3. Editable models, delegate <-> view

1. Binding loops!

2. Common solution: implement in C++

3. Signals that returns the value when its changed etc

4. Solution: synchronizer (6.10)

1. Works with any property

2. `Synchronizer on value ( property alias source: delegate.value )`

5. Having models updated wont update the view data

1. Editing affects only internal model

6. QRangeModel

7. SortFilterProxyModel

1. Tech preview

8. 6.10 future work

1. finish SortFilterProxyModel

2. Shorthand syntax for Synchronizer

3. QML keyword for sticky bindings

4. Properly integrate QRangeModel

5. 'Transactions' for Synchronizer (async?)

9. Tableview delegate and treeview delegate are coming!

1. Hoping for increased adoption
 2. 50 million tonnes of e-waste generated annually!
 3. Big CTOs need solutions, not OS
 4. There is demand for linux / OSS
 5. Need to make things with
 6. - little learning
 7. - minimal distractions
 8. Not competition between distros/DEs
- example windows system update costs can be 683350€
10. for around 1000 systems
 11. Moving to linux is of course a risk
 12. - Needs 3rd level support
 13. example: Moving from windows to linux
 14. ~ 62500 € cheaper due to less license cost
 15. Technically we can save up to
 16. 2.7 billion eur
 17. Having support license that helps keep
 18. things running could be way to it
 19. win: 1 eng / 50-100 desktops
 20. Linux: 1 eng / 3000 desktops
 21. Determine groups
 22. Work with mgmt
 23. Discover hardware
 24. start with limited trial
 25. No gamer support, lol
 26. Build images, not distros
 27. - K1W1-NG cli tool
 28. - /etc/skel
 29. - AD is your friend
 30. - Backups!
 31. 1-25 pc: USB stick
 32. USB Hard stick
 33. 1-100 pc: desktop / remote
 34. - KVM
 35. 100+ PC: automation, like ansible, "autogast"
 36. Network boot, wake on lan
 37. RDP!
 - 38.
 39. Pick your battles
 40. Reduce initial scope
 41. Use images
 42. automate installs
 43. Minimise user complaint opportunities!
 - 44.

1. World is doing bad...
2. But KDE is doing ok!
3. In chaos, trends that benefit KDE
4. ~ Big tech losing appeal
5. ~ EU onshoring
6. ~ cloud, AI, WM skepticism
7. Possibilities
8. ~ Plasma preinstall
9. ~ Plasma in companies / institutions
10. ~ Maybe even 15% marketshare?
11. ~ KDE jobs?
12. Whats the catch?
13. ~ We need to be ready
14. ~ Prepare solutions, Ready ones
15. What does ready look like?
16. ↓
17. Known → Presentable → Easy to get → Works
18. → TCO and support
- 19.

20. How is plasma doing? Nate opinion?

21. Known: good word of mouth ↑

22. nerd bubble ↓

23. Little ads ↓

24. Presentable: professional ↑

25. Easy to get: No dl here button ↓

26. some preinstalls ~

27. Works: easy, featureful, adaptable ↑

28. ~ reliable

29. TCO support:

30. No license costs ↑

31. commercial Lts ↑

32. Can pay for tailoring ↑

33.

34. Plasma weaknesses:

35. ~ Not known

36. ~ Not easy to get

37. Fragmented story!

38.

39. Positive donation comments! ♥

40.

41. Users love us! ☺

42.

43. Big picture to-dos!

44. Important:

45. Advertise out of bubble!

46. Clear getting method!

47. Accessibility!

48. pre install readiness!

49. Less important:

50. Better aesthetics?

More features?

Redesign website?

customizability?

More guerilla marketing?

More awareness! Ease of adoption!

↳ presence, ads, promo ↳ KDE Linux

↳ "youtube friendly" ↳ Accessibility!

Details should advance big picture!

↳ Long term utility

↳ Dont waste time on unnecessary stuff

Who can use this > pixel margins

Window of opportunity will close!

Our software is used by at least millions
~ We dont really know

Awareness	Presentation
Adoption	Ease of use

Very optimistic ☺

1. Artists are moving away from Windows

2. Propt. sw is getting unpopular

3.

4. Why KDE?

5. ~ We look like wind

6. ~ Krita part of KDE

7.

8. MUSCLE KONGA

9.

10. Plasma has infra for tablet input

11. ~ config for graphics tablet

12.

13. Our KCM was complicated but less
14. stuff than official sw

15.

16. New KCM looks good and
17. has more settings

18. ~ Lots of changes since P5

19.

20. Things are better than on X11

21.

22. Convincing people to switch:

23. ~ Lots of "social" debt

24. ~ Bad rep for WL

25. ~ Try again!

26. ~ Downplay wayland part!

27. ~ Communication is key

28. Artists have been switching!

29. Non-Foss folk asking WL questions

30.

31. Some HW feats are missing

32. Krita WL slow progress for native WL

33. Lacking driver support (kernel)

34. ~ Biggest blocker

35. ~ some tablets have compatibility mode, not good

36. All DE's interested in better drivers!

37. Fixing in userspace is troublesome

38. eBPF way forward?

39. Ecosystem work needed!

40. TODO:

41. Funding, vendor coop, drivers, port sw to linux
42. more marketing/visibility.

43. # Made On KDE

44. Don't be foss>bro

45. Don't use debian lol

46. Test and report bugs

47.