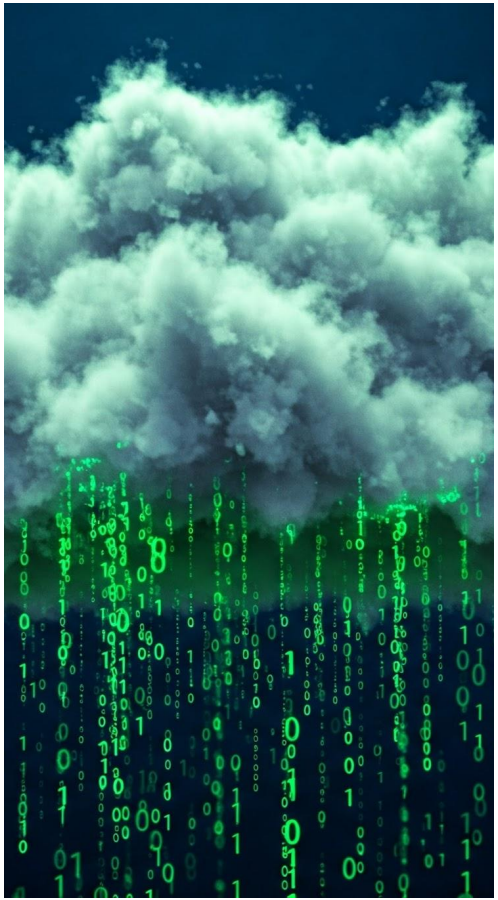
OpenStack in the Real World

Jay Faulkner (JayF)
G-Research Open Source Software

Who am I?

- OSS Dev @ G-Research Open Source Software since 2022
- Host of **GR-OSS OUT** podcast
- OpenStack operator/developer for over a decade
- Launched first OpenStack Bare Metal Public Cloud
- Worked on/architected private clouds at multiple companies
- Gentoo contributor

Clouds aren’t real

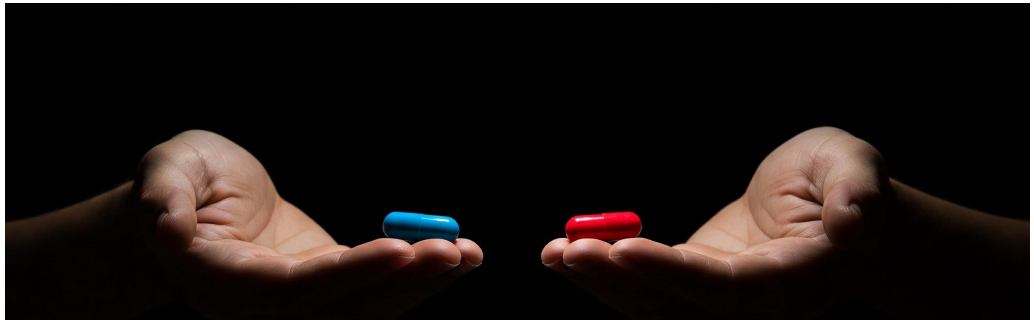


Cloud

You can stay in your comfortable workflow; calling APIs to get your infrastructure, unencumbered by physical realities…

Real World

…or you can come with me and see the details hiding behind the surface. The unpretty reality that powers **The Cloud**.



Welcome to the Real World

Real World: Purchasing

Lots of things to consider:

- Does the server have a standards-compliant BMC?
- Any additional license charges for
  - Redfish?
  - Virtual Media support?
- Do you care about network?
  - Get devices that can be automated
- How many servers?
  - Capacity planning!
  - **Excess Capacity == more uptime**

Now you have an idea what servers you need…

Real World: Onlining a new server rack (physical)

…time to bring them online!

What do we need?

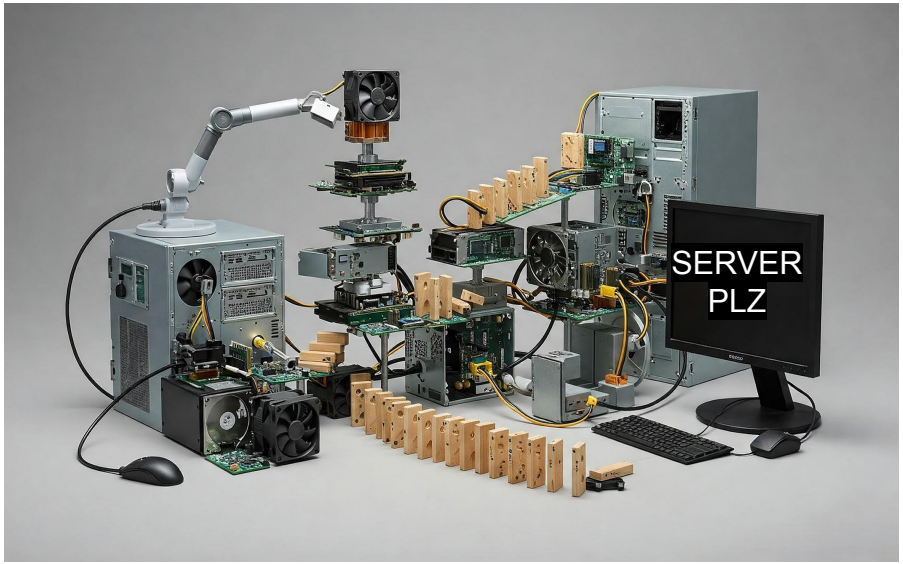
- List of BMC MACs & logins
- Server correctly cabled to rack switches
- Network configured & uplinked to switch
- Power run to rack PDU
- Power run to each server
- Hardware works first time
- All hardware delivered accurately

Only after all this can be trusted do you can start getting cloud software involved.

Real World: Going from online -> provisioned (Ironic)

Ironic needs a lot of things to go right:

- Ironic node creation
  (manually, automated, or inspected)
- Conductor has network access to BMC
- Conductor can configure network
- Conductor access to provisioning network
- Agent image is accessible and **valid** for hardware
- Agent can reach the Ironic API
- Instance image is accessible and **valid** for hardware use
- **BMC actually turns on and boots server from disk**



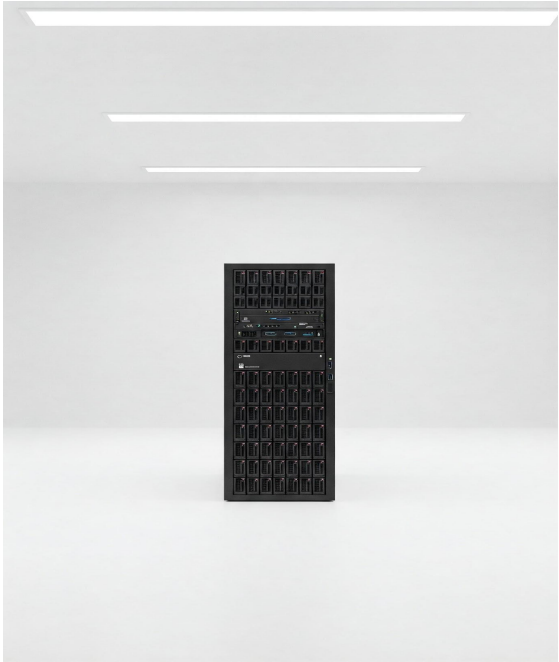
Real World: Do this a lot, over and over again

It has to work

every time

…our users don’t want
to care about that!

It’s the job of a cloud to ensure they don’t have to.



Cloud isn’t just APIs and servers; it’s **a service**

…and you’re competing to provide that service!

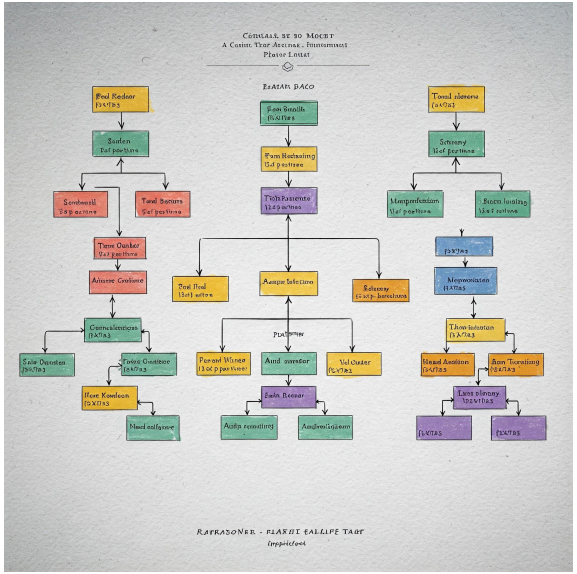
Who are your competitors?

- Public cloud providers
  - Shadow IT?
- Other internal infrastructure teams
  - “Traditional” server deployment
  - VMWare
  - That internal system that’s been in use for 30 years
- A fantasy sold by a technology salesperson
  - The real world rarely holds up to fantastical claims made by commercial platforms.
  - It’s likely you’ll be measured against them anyway.

Part of our job is to show (and ensure!) the private, internal cloud is the right choice.

What is involved in building a quality service?

- Product Management
- Marketing & Sales
- Customer Support
- Service Level Agreements
- Billing / Pricing



Let’s compare what these concepts mean in public vs internal clouds.

Public: Product Management

- Proactively researching into new technologies or approaches customers may want
- Constantly telling customers about new features they could utilize
- Focusing investments into initiatives with good value:effort ratio.

Private: Requirement Negotiation

- Documenting the basic set of business requirements for your cloud.
- Figuring out how to prioritize future work.
- **Ensuring your budget matches the business requirements**

Public: Marketing/Sales

- Constantly telling customers about new features they could utilize
- Engaging customers by showing the power of workflows they can enable
- Heavy focus on onboarding/migrations; including quick start guides and basic how-tos

Private: Internal Comms

- Telling users about new features or good practices for use
- Ensuring successes are as visible as failures
- Ensuring KPIs should are 100% under the control of the cloud team

Public: Customer Support

- Often handled in private, documented interactions; only insight customers have into failures are their own
- Failures of large enough scope can cost years of credibility
- Lots of public self-service support resources

Private: User Support

- May be handled in a more visible way to others (e.g. a public chat or mailing list) – failures are extremely visible
- Important to be professional and polite. A single bad interaction at a critical moment can cost years of credibility
- Self-serve resources for users to understand and prevent failure

Public: SLAs

- Lists acceptable uses of the service
- Defines exactly what you're paying for
- Sets maintenance windows
- Service uptime guarantees

Private: Expectation Management

- Define usage patterns for the service
- Clear scope for what each team is responsible for.
- Document availability expectations:
  - APIs may not need/have 24/7 uptime
  - Users may be responsible for failover between AZs
- Policies for support ticket turnaround time and on-call

Public: Billing / Pricing

- Direct relationship between value and charge
- Accuracy and formal processes extremely important to avoid under/overcharge
- Can sometimes be used to encourage migrations off older technology

Private: Chargeback

- Creates a relationship between value provided and received
- Accuracy is usually less important; may be enforced informally
- Provides a mechanism to ensure the most deserving projects get hardware when capacity is tight

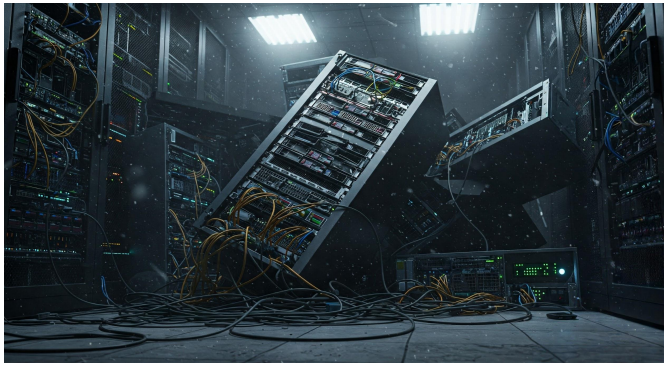
…but public clouds don’t hold the upper hand

Internal Cloud Advantages

Advantage:
Know the business

Know the business: Capacity Management

- Unused hardware can be used for internal projects opportunistically for minimal additional cost.
- Capacity headroom and flexibility can create more consistency (via retries) for users.
- Upgrade hardware on the business' timeline

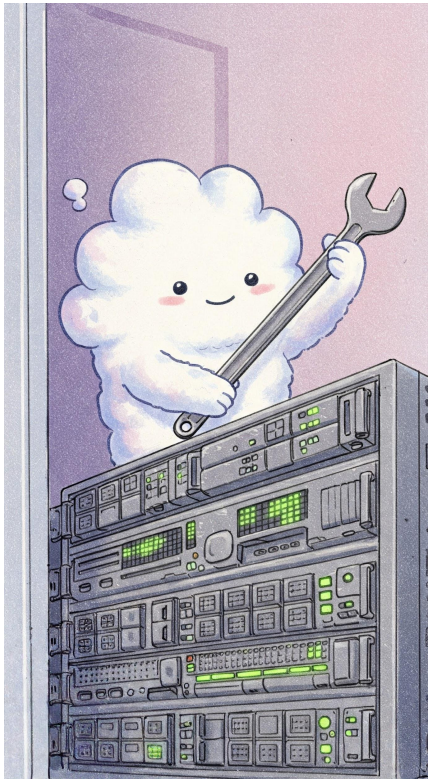


Know the business: Managed Service

- Offer levels of managed services or instance management

- Managed services like:
  - Swift for object storage
  - Trove for easy DB provisioning
  - Magnum for K8s provisioning

- More scope == more value
  - Backups
  - Monitoring
  - Firmware Updates
  - OS Updates



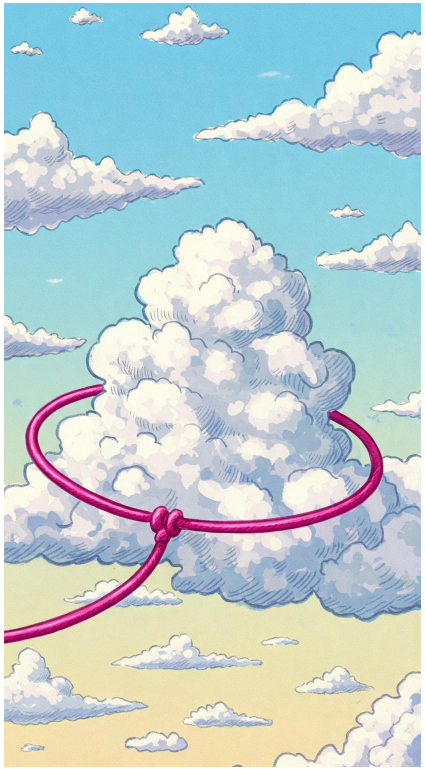
Know the business: Advanced troubleshooting

- Unexpected performance characteristics?
  Isolate the software for benchmarking!

- Driver or Firmware update impactful?
  Track it down or roll it back!

- Hardware failed or inadequate?
  Maybe recoverable or upgradable!

Advantage:
Control your data



Control your data: Security

- Completely eliminate risk of cross-tenant data leakage
- Ensure your data is used for your business
- Reduced external attack surface

Control your data: Locality

- Avoid the data gravity lock-in
- Cross fewer political borders for data sovereignty and compliance
- Direct control over DR and backups

Advantage: **Optimization**

Optimization: Tailored Environment

- AI accelerators or GPUs to run AI models
- Controlling network latency with advanced networking
- Industry-specific or other specialized hardware
- Maintenance windows that fit your business/timezone

Optimization: Business Integrations

- Use existing automation alongside the cloud

- Automatically update internal systems; e.g. inventory or finance

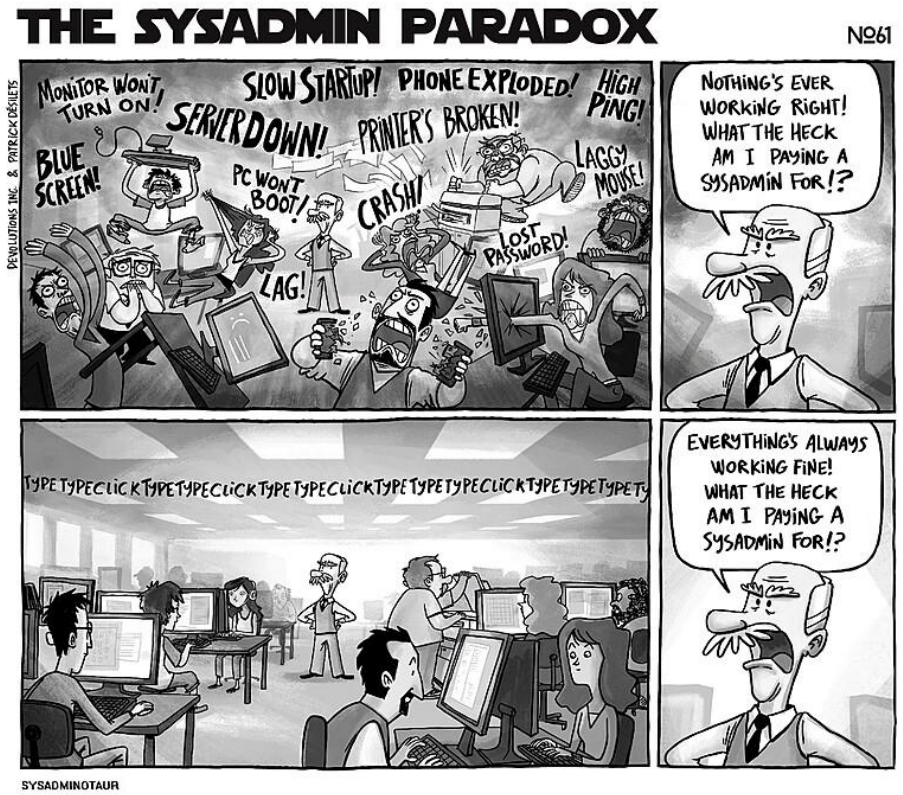
What can I do today?

Action: Represent your users

- Meet with your users; find their pain points
- Propose plans to management to reduce these pain points
- Document the outcome

Action: Represent yourself

- Document the scope of the cloud team
- Rework dashboards to use positive metrics that represent team scope
- **Be loud when the cloud does good things.**



Action: Technology / Business partnerships

- Engage the planning process for future initiatives
- Ensure leaders understand the relationship between your budget and trade offs
- Check: does your usage match up with the value those users provide?
- Monitor and correct perceptions about the cloud: **Does the CxO think your cloud team is doing a good job?**

Conclusions

Running a Cloud involves much more than OpenStack

Providing value is more important than technological dogma

Tailor your cloud to business needs

Document team scope; then exceed it

Keep your users in the Matrix.

The real world is cruel.

Questions?

Check out the GR-OSS OUT podcast!

https://youtube.com/@oss-gr

https://podcast.gr-oss.io

…or search your podcast app of choice for “GR-OSS”. Next episode featuring fungi and clarkb will be out this week!

Want to work on/with OSS in Dallas, TX or London, UK?