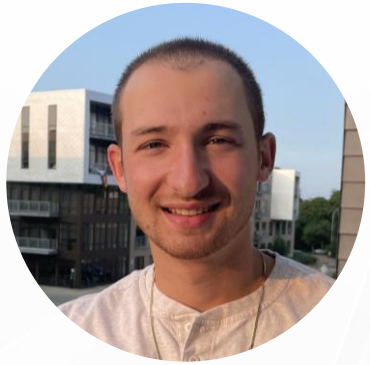
Open@Epic CONFERENCE

Going the Distance with API Performance

Designing and implementing FHIR apps that scale

Presenters



Danny Pocklington

EDI

Epic



Jeff Minnerly

Performance TS

Epic

Key Observations



Data exchange is growing



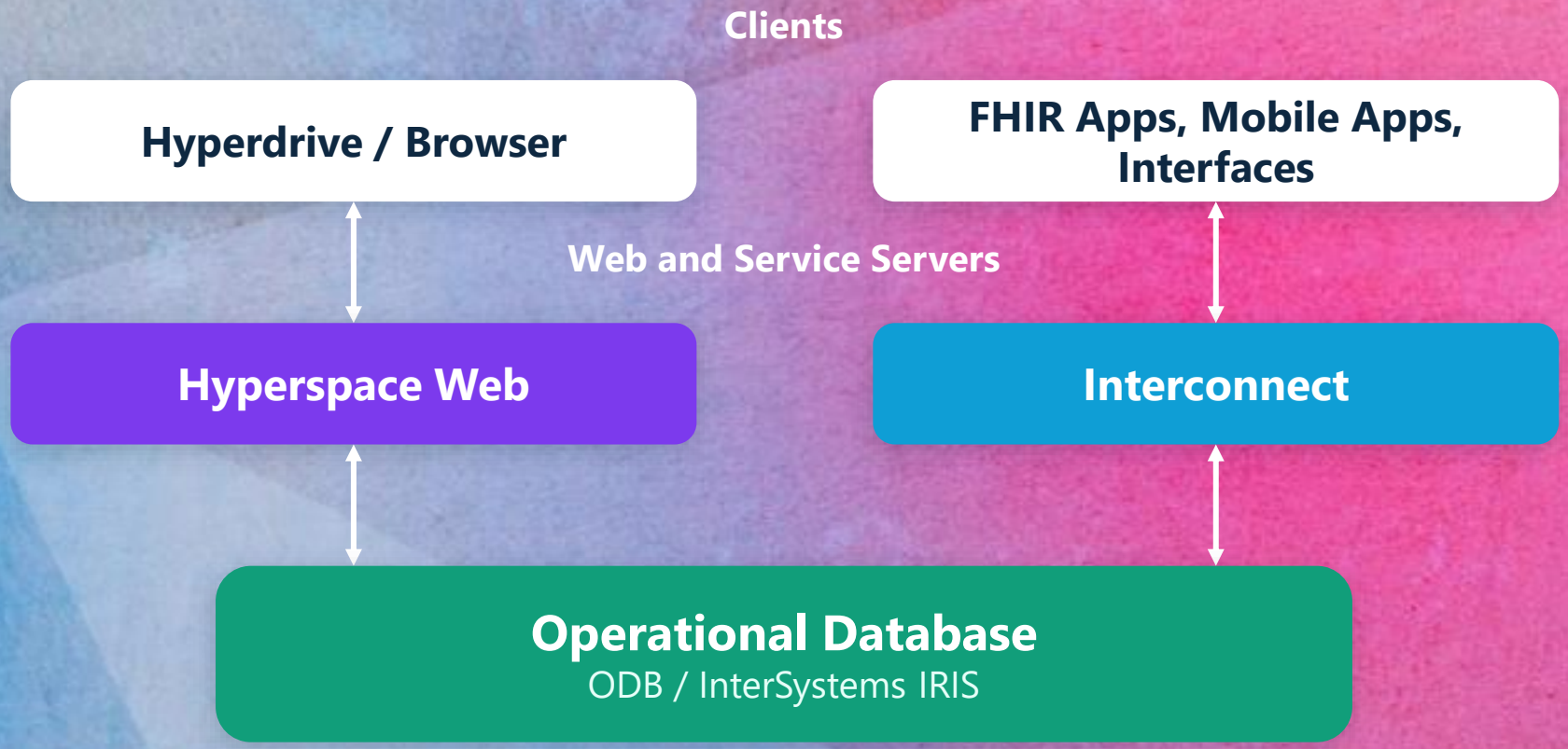
Scaling data exchange is hard



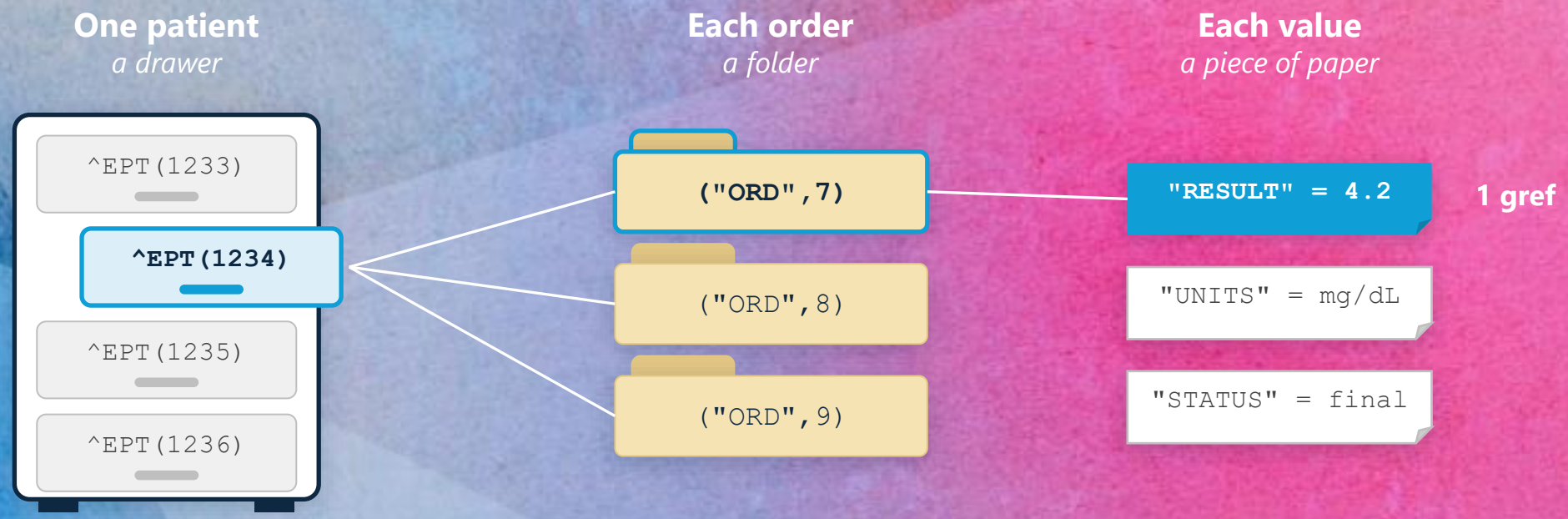
There are strategies to scale data exchange efficiently

Key Concepts

One Shared Database



What is a Global Reference?



A global reference is one read or one write of one of these nodes.

All Requests Are Not the Same

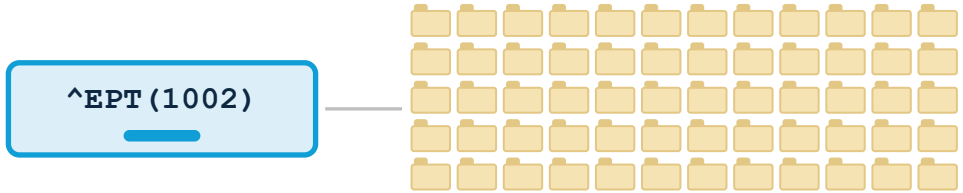
Same request `GET /MedicationRequest?patient={id}`

Patient A
1 order



3 grefs
1 order × 3 values

Patient B
500 orders



1,500 grefs
500 orders × 3 values

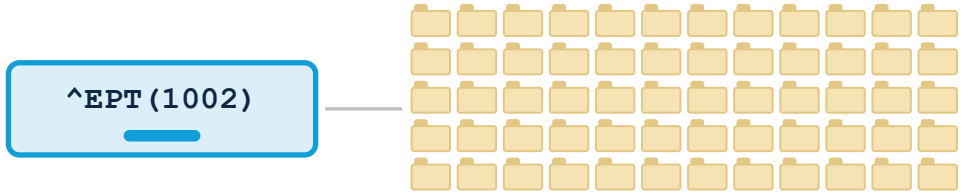
Same request. 500 times the work.

Your Query Determines the Work

Same patient 500 orders over 20 years

Full history

`GET /MedicationRequest?patient={id}`



1,500 grefs

500 orders × 3 values

Last month

`GET /MedicationRequest?patient={id}&authoredon=ge2026-09-01`



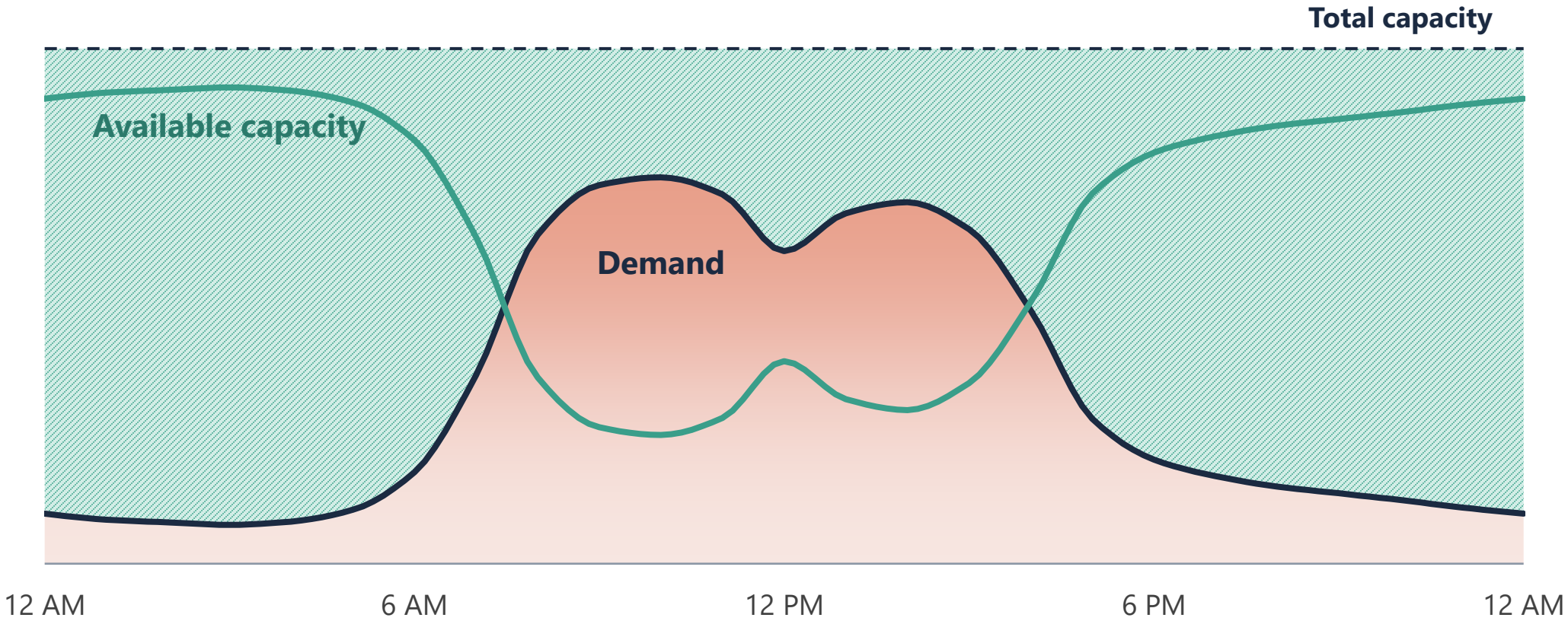
75 grefs

25 orders × 3 values

Both are one call. One does 20 times the work.

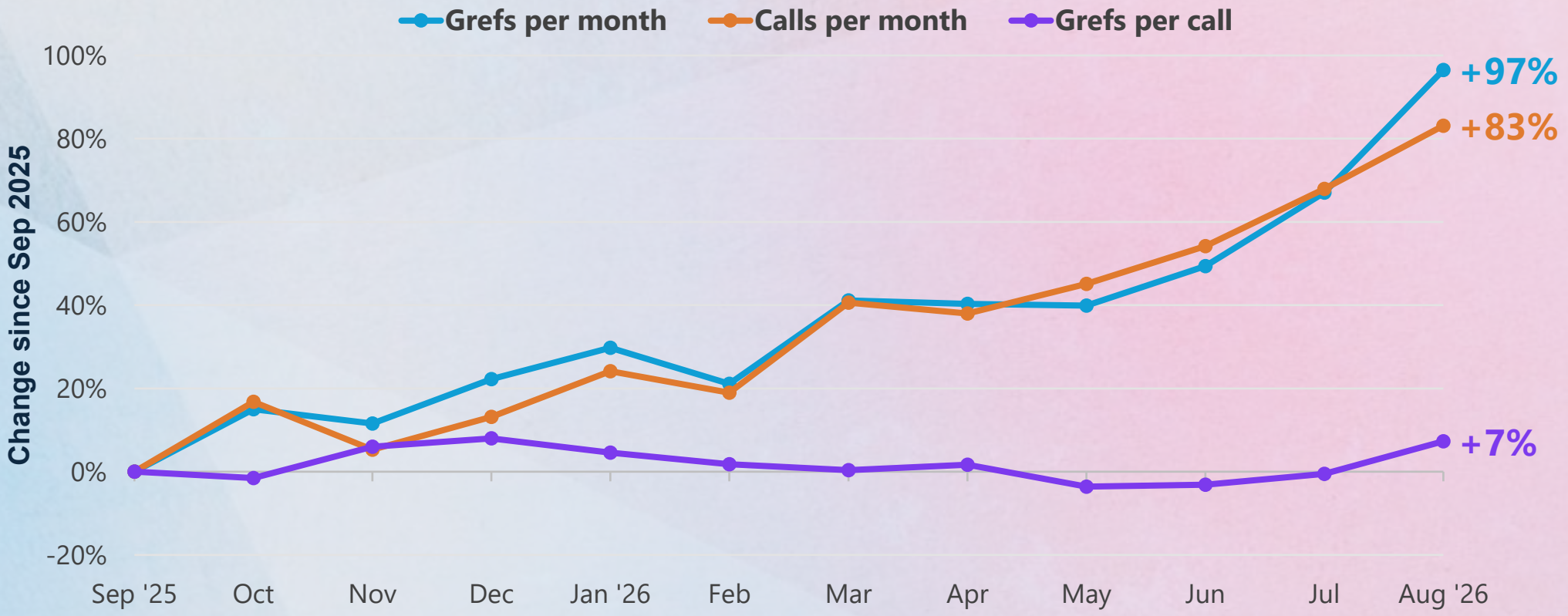
Illustrative numbers. Actual savings depend on the resource, the parameter, and the chart.

Peak Hours


Total capacity
Available capacity
Demand
12 AM
6 AM
12 PM
6 PM
12 AM

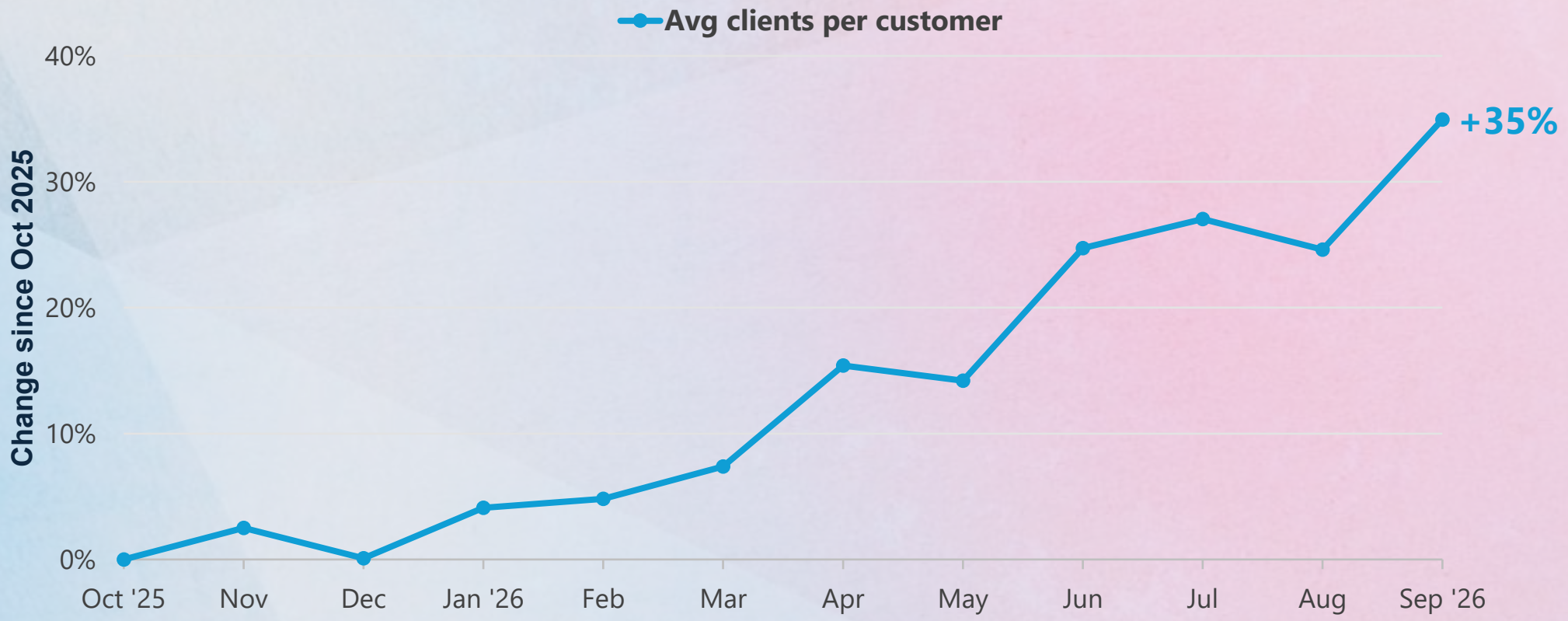
Trends in Data Exchange

FHIR Performance Trends



Full-day FHIR traffic across Epic health systems, Sep 2025-Aug 2026. Monthly totals.

Client Integration Growth



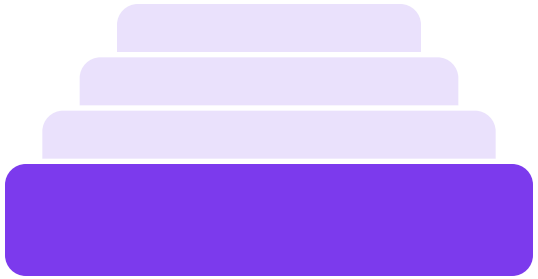
Named API clients per customer, averaged across 626 customers (OID), by month.

Data Exchange Patterns

Full History



Full Chart



Many Patients



What is Epic doing to help performance?

Core focuses right now



Protecting Hyperspace processes

Keeping clinician workflows fast when API traffic spikes



Making APIs more efficient

Doing less database work for every API request

Why Should You Care?

The Memory Squeeze

The cost of RAM is up about 600%.



PRICE PER GRAM

Server RAM vs. gold

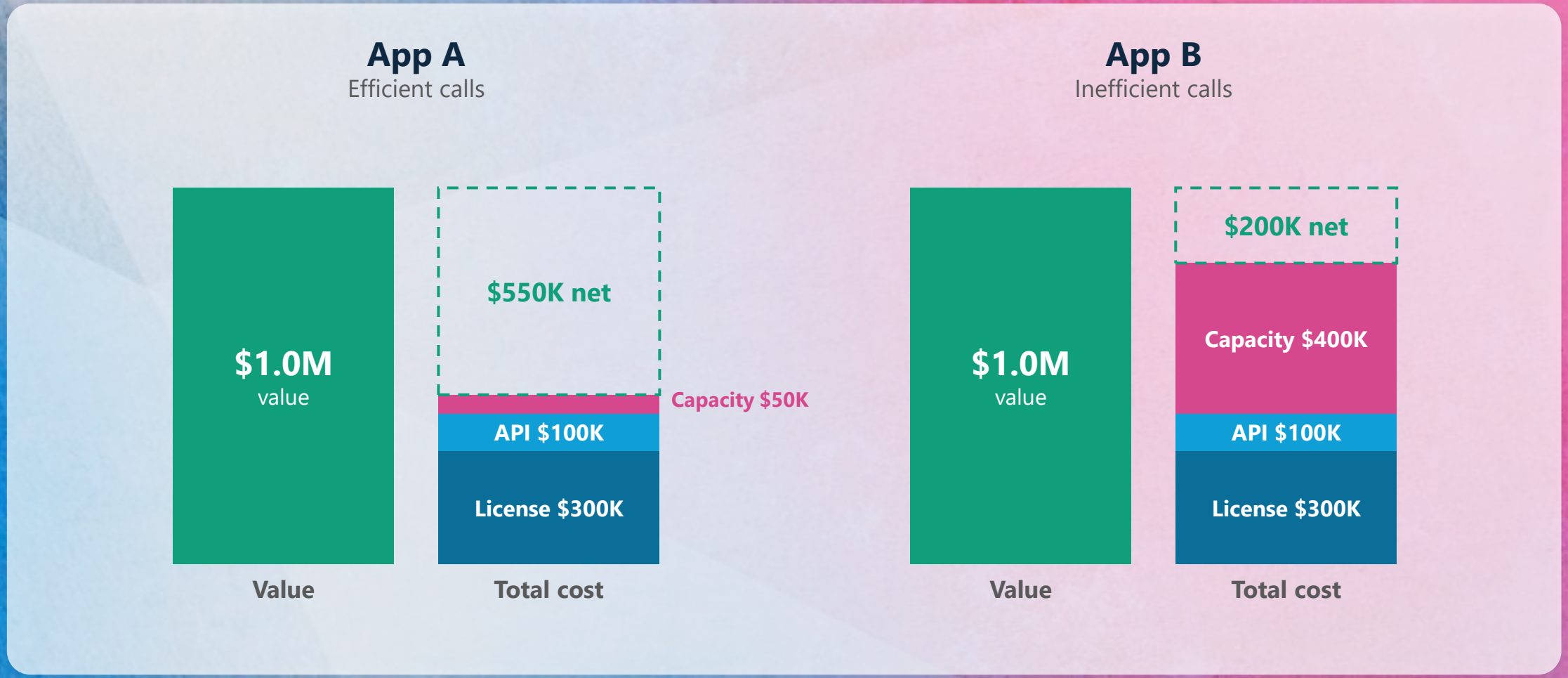
Server RAM	Gold, 24k spot
$164.58	**$139.61**
per gram	per gram
▲ 17.9% above gold	

Server RAM **$164.58 / g**

Gold, 24k spot **$139.61 / g**

64 GB DDR5-6400 RDIMM, $3,950 ÷ 24 g · Gold: JM Bullion 24k spot, Sept 22, 2026

Efficiency Sells



Same features. Same price. The efficient app is the easier yes.

When Capacity Runs Out

Interconnect throttles API traffic

↓

Critical integrations and jobs slow down, then stop

↓

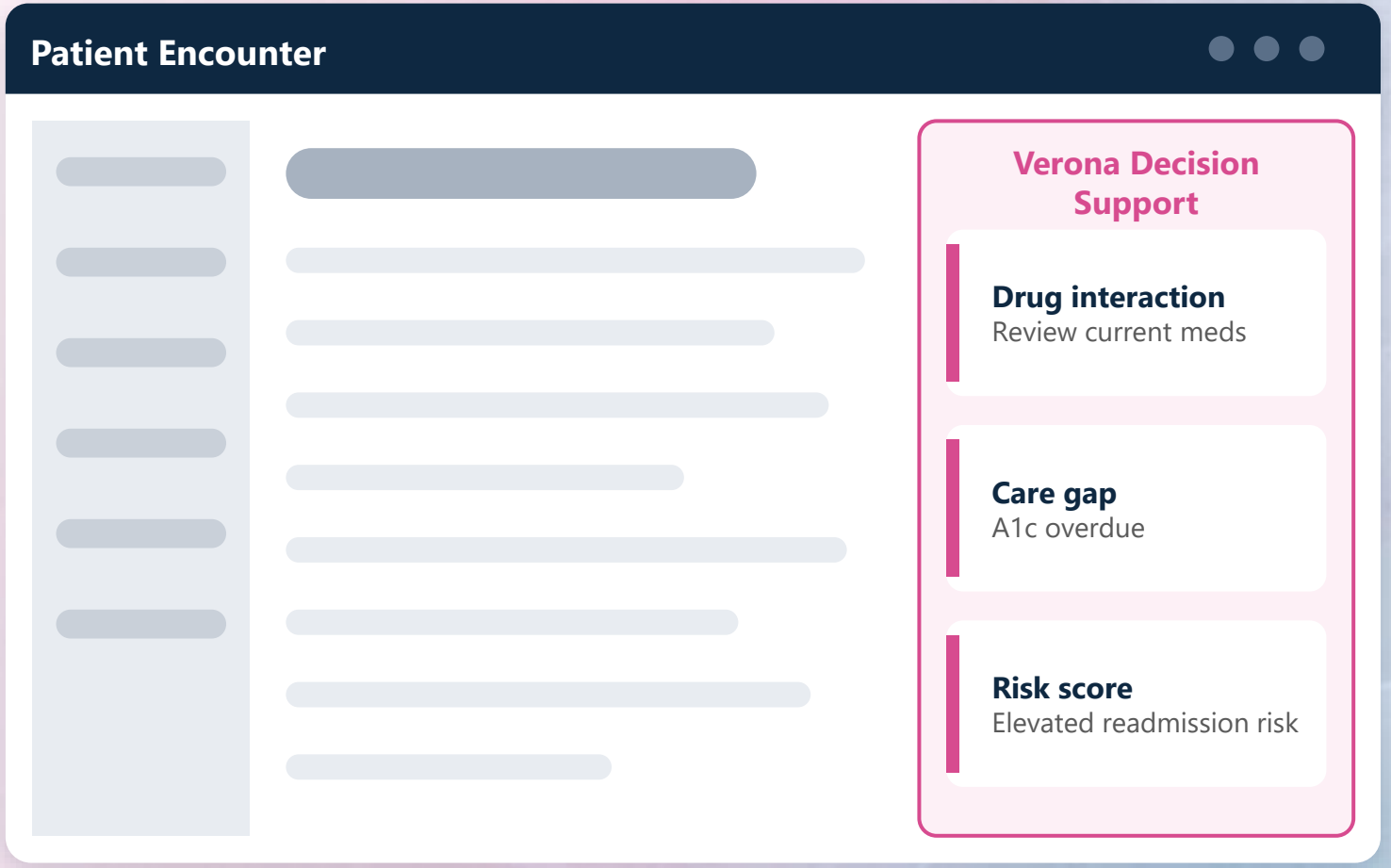
Response times slow for every user

↓

Identifying the source takes cross-team analysis

Generalized Example

Verona Decision Support



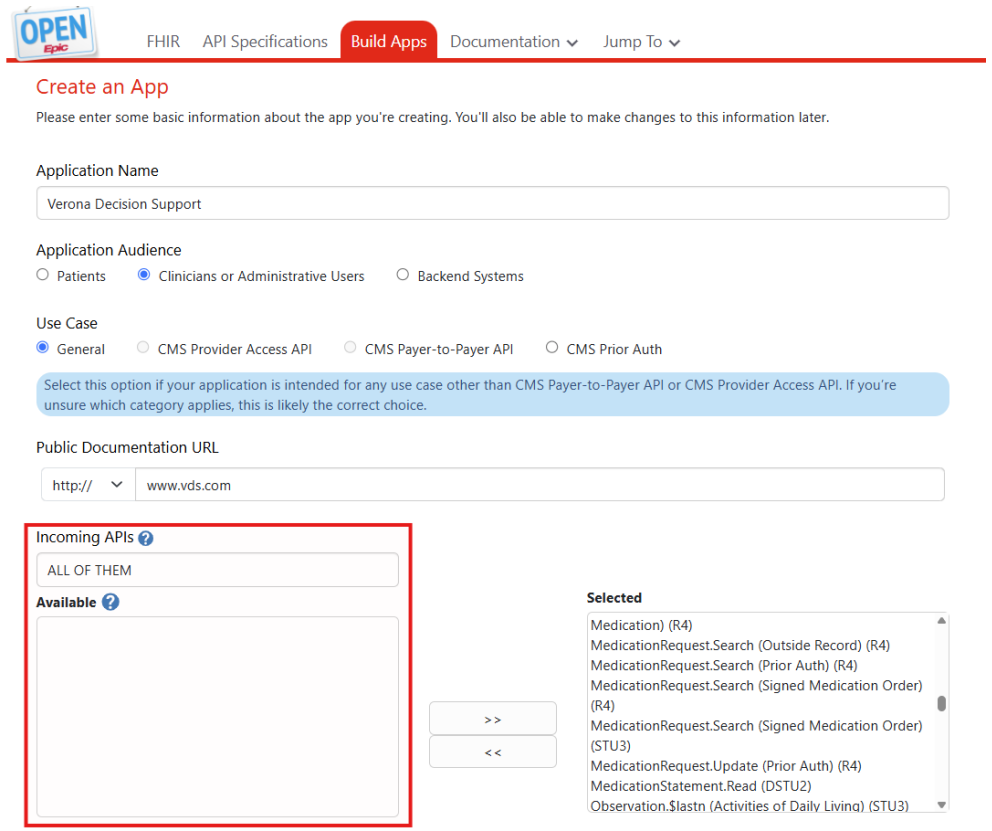
01 **Decision support**
Clinical guidance at the point of care

02 **Leverages historical data**
Opens alongside every patient encounter

03 **Embedded in Epic**
Lives in the sidebar of the clinician's workflow

Data Exchange Design

Scoped every API



Pulls everything, unbounded

Clinician opens an encounter

↓

App launches

↓

Call all APIs for Patient

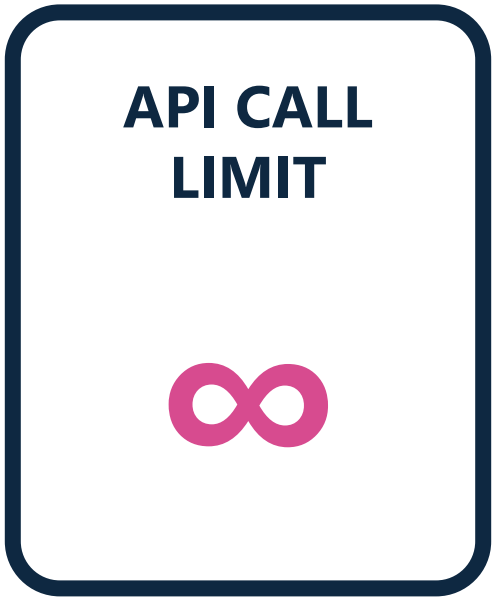
↓

No date range · No type, code, or category

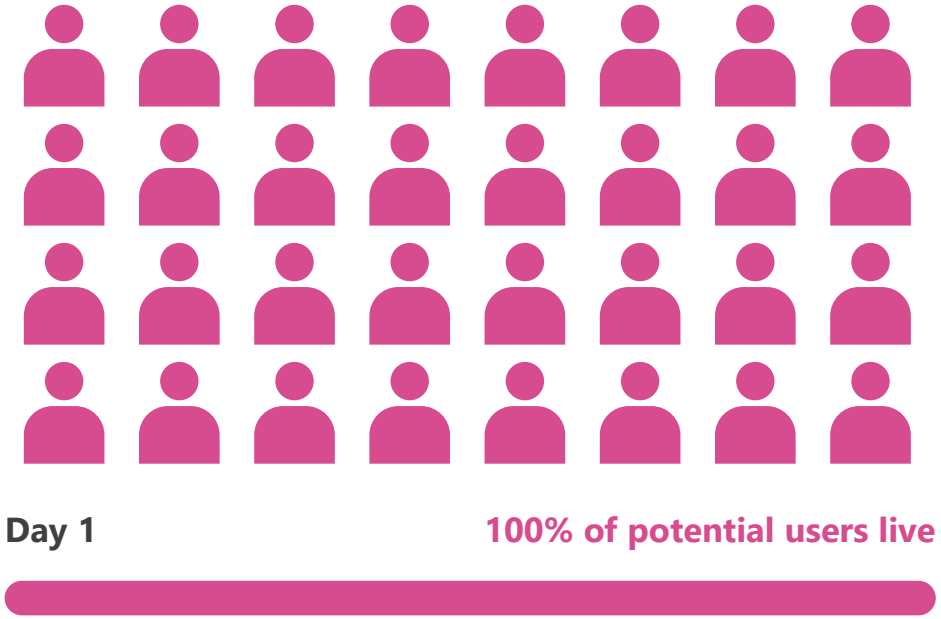
Rolling Out the App

How was the app rolled out?

01 No limit on API calls



02 Every user on day one

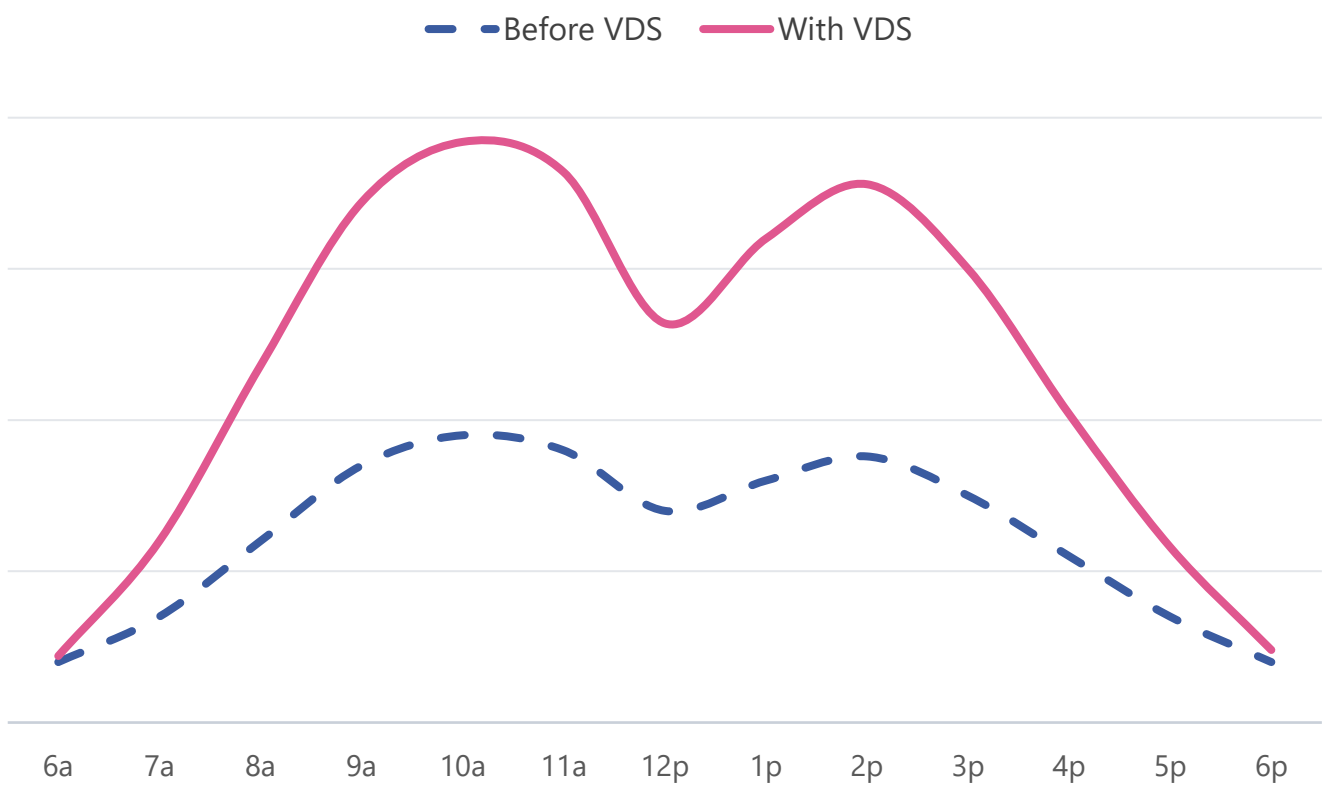


The Impact

What happened when it went live?

GREFs/sec across a clinic day

Illustrative



2× **GREFs/sec** during peak hours

Load scales with encounter volume, so it lands right on top of the busiest hours

CPU utilization



⚠ **System starts protecting itself**

What is the Effect?

Nobody knows yet that it is the new app

Interconnect throttles API traffic

↓

Critical integrations and jobs slow down, then stop

↓

Response times slow for every user

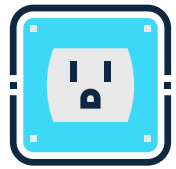
↓

Identifying the source takes cross-team analysis

Real World Data



5,853



times Interconnect has had to reduce connections in the last year

Designing an Efficient App

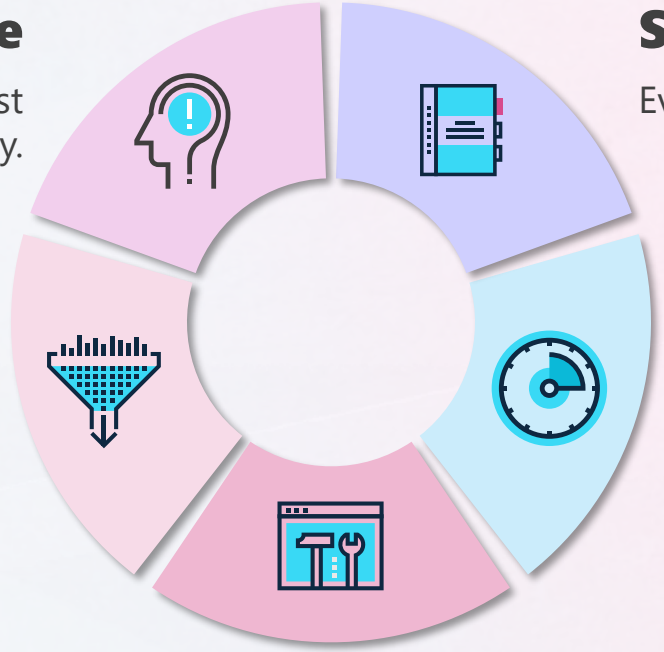
Designing an Efficient Application

Ask for Data Once

Pull what changed since your last call, not the full history.

Spread the load

Even calls out across the day.



Ask for less

Filter by date, type, and patient so you get only what you use.

Time your calls well

Make calls when the least work is being done on the database.

Use the right tool

Choose the exchange method that fits the job, not the one you know best.

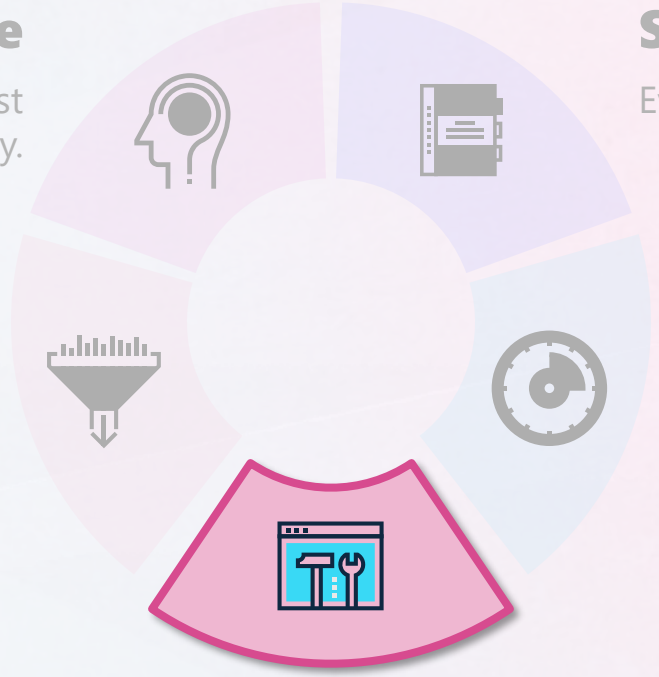
Designing an Efficient Application

Ask for Data Once

Pull what changed since your last call, not the full history.

Spread the load

Even calls out across the day.



Ask for less

Filter by date, type, and patient so you get only what you use.

Time your calls well

Make calls when the least work is being done on the database.

Use the right tool

Choose the exchange method that fits the job, not the one you know best.

Data Exchange Toolbox

RESTful FHIR

Bulk FHIR

Event Driven Interfaces

Epic Non-FHIR APIs

Extracts

CDS Hooks

Redesigning Verona Decision Support

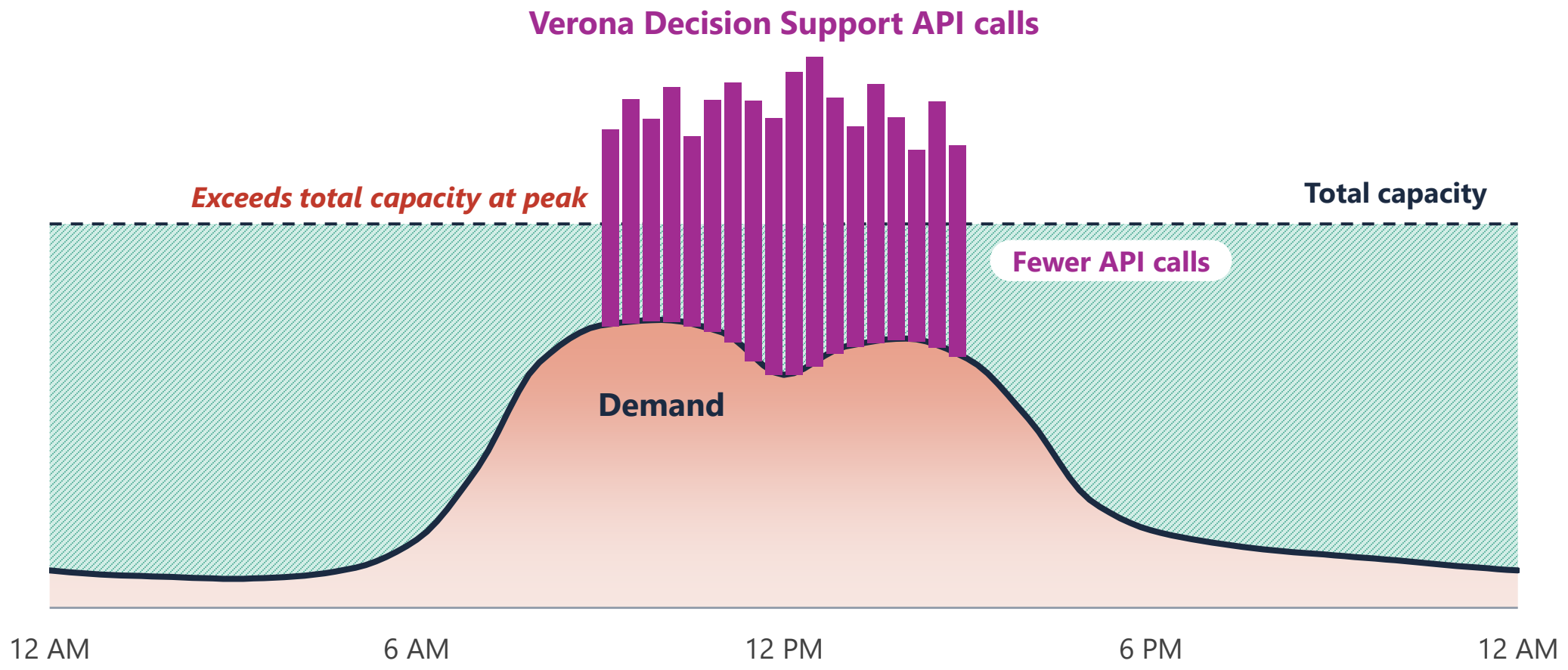
RESTful FHIR

- Only call relevant APIs
- Filter to what changed since the last extract

Extracts

- Pull full patient history one time with large extract
- Runs in the quiet hours, not during clinic
- Load it once, reuse it every encounter

Fewer Calls at Peak



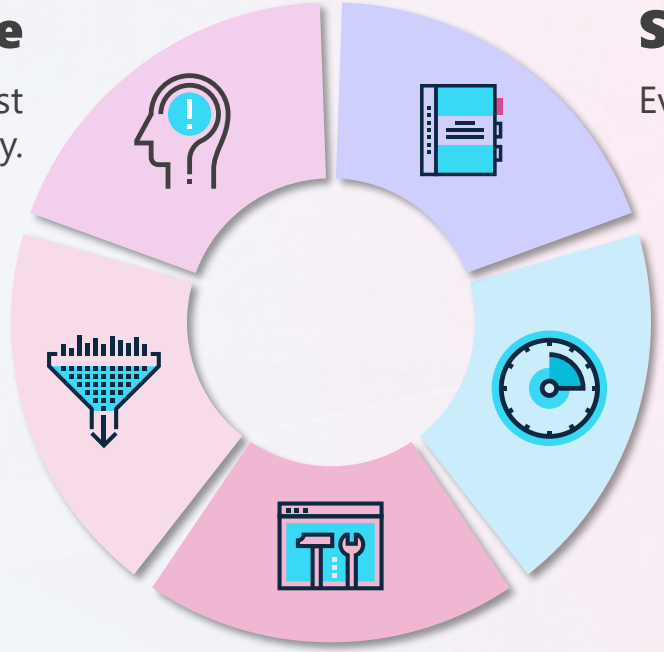
Designing an Efficient Application

Ask for Data Once

Pull what changed since your last call, not the full history.

Spread the load

Even calls out across the day.



Ask for less

Filter by date, type, and patient so you get only what you use.

Time your calls well

Make calls when the least work is being done on the database.

Use the right tool

Choose the exchange method that fits the job, not the one you know best.

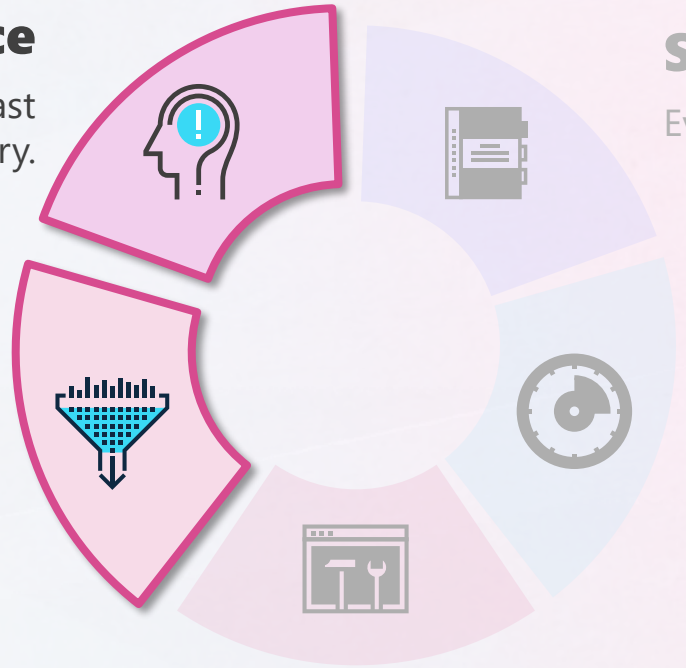
Designing an Efficient Application

Ask for Data Once

Pull what changed since your last call, not the full history.

Ask for less

Filter by date, type, and patient so you get only what you use.



Spread the load

Even calls out across the day.

Time your calls well

Make calls when the least work is being done on the database.

Use the right tool

Choose the exchange method that fits the job, not the one you know best.

Asking for Less

01 History comes from the extract, so FHIR only asks for updates




```
GET /Observation?patient=123&_lastUpdated=gt{last encounter}
GET /DocumentReference?patient=123&date=ge{last encounter}
```

02 Category values narrow the search

```
GET /Observation?patient=123&category=laboratory
```

✓ **Laboratory** | Vital signs | Social history | Survey | Exam | Imaging | Procedure

Now Supported: _lastUpdated

NEW

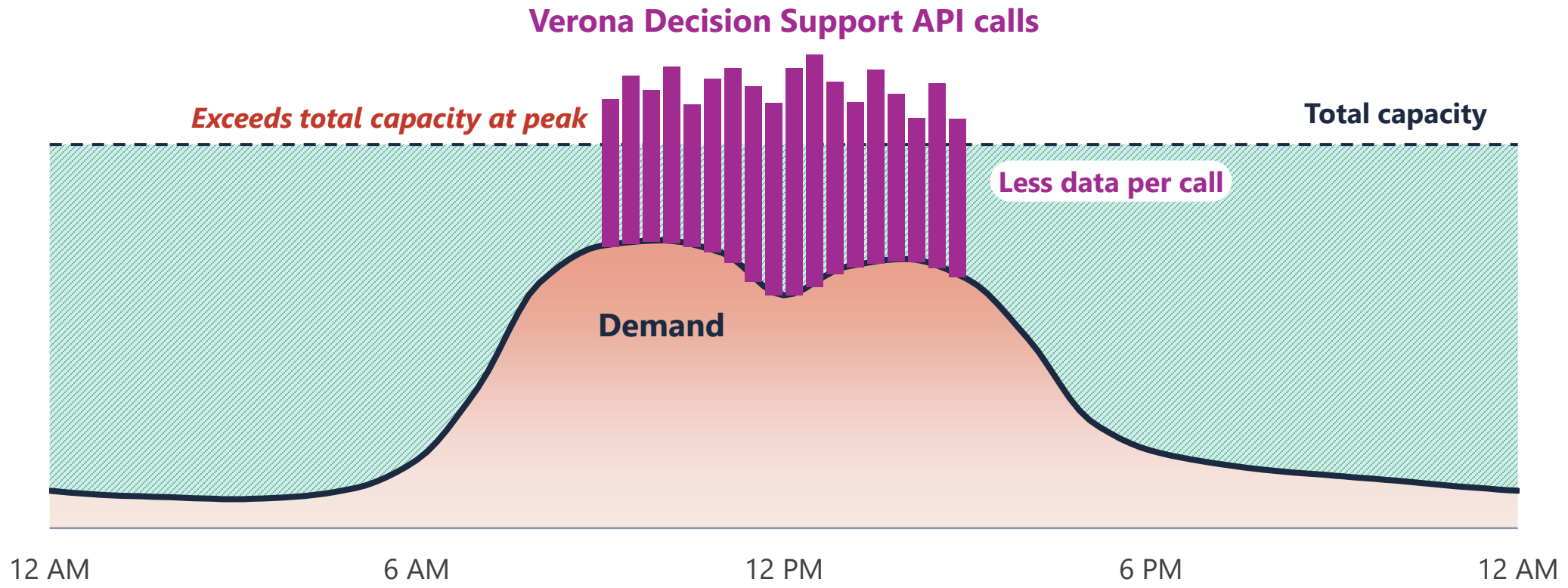
Filter by `_lastUpdated` on

- ✓ **`DiagnosticReport.search`** (Results)
- ✓ **`Observation.search`** (Labs)

```
GET /Observation?patient=123&_lastUpdated=gt{last encounter}
```

Ask only for what changed since your last call.

Less Data per Call



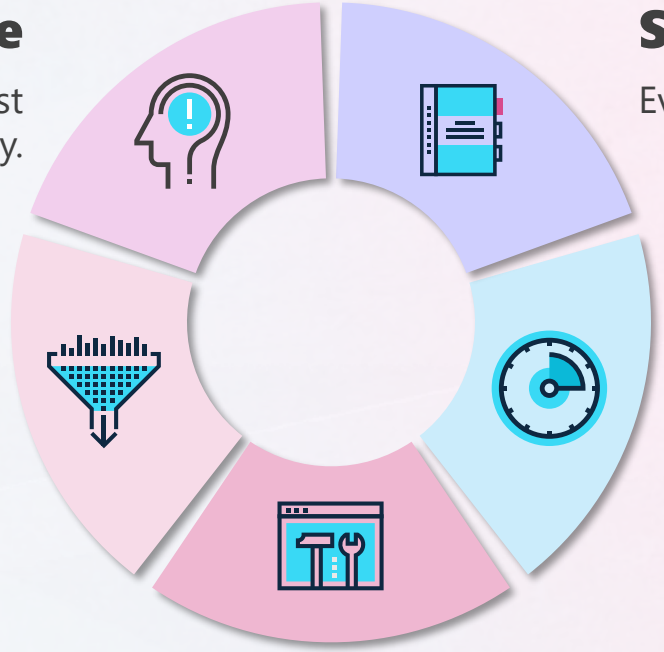
Designing an Efficient Application

Ask for Data Once

Pull what changed since your last call, not the full history.

Spread the load

Even calls out across the day.



Ask for less

Filter by date, type, and patient so you get only what you use.

Time your calls well

Make calls when the least work is being done on the database.

Use the right tool

Choose the exchange method that fits the job, not the one you know best.

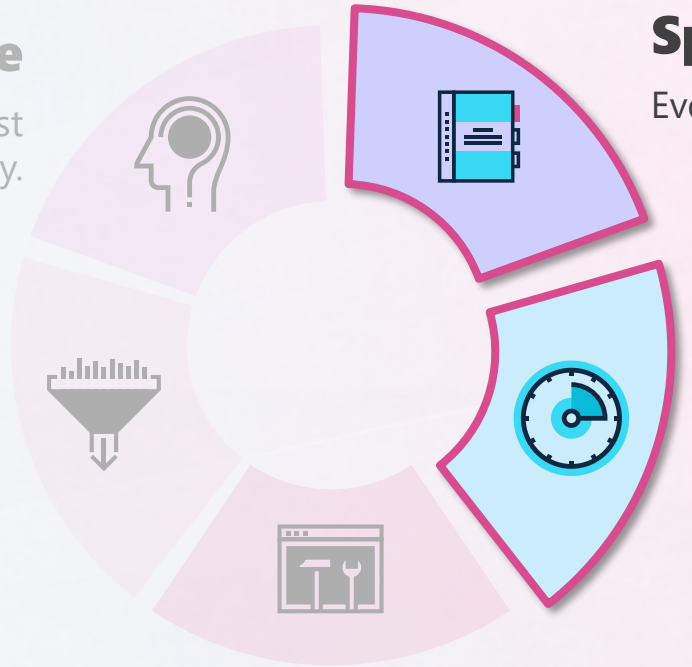
Designing an Efficient Application

Ask for Data Once

Pull what changed since your last call, not the full history.

Spread the load

Even calls out across the day.



Ask for less

Filter by date, type, and patient so you get only what you use.

Time your calls well

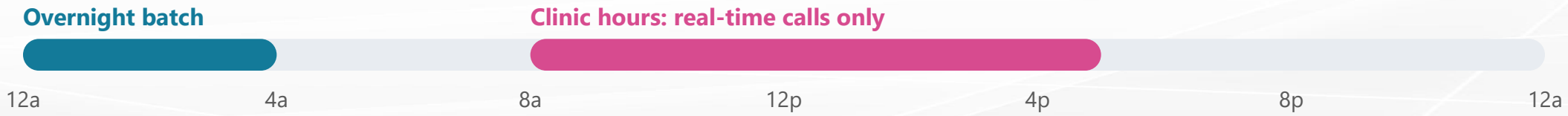
Make calls when the least work is being done on the database.

Use the right tool

Choose the exchange method that fits the job, not the one you know best.

Moving Work Overnight

A day of VDS calls



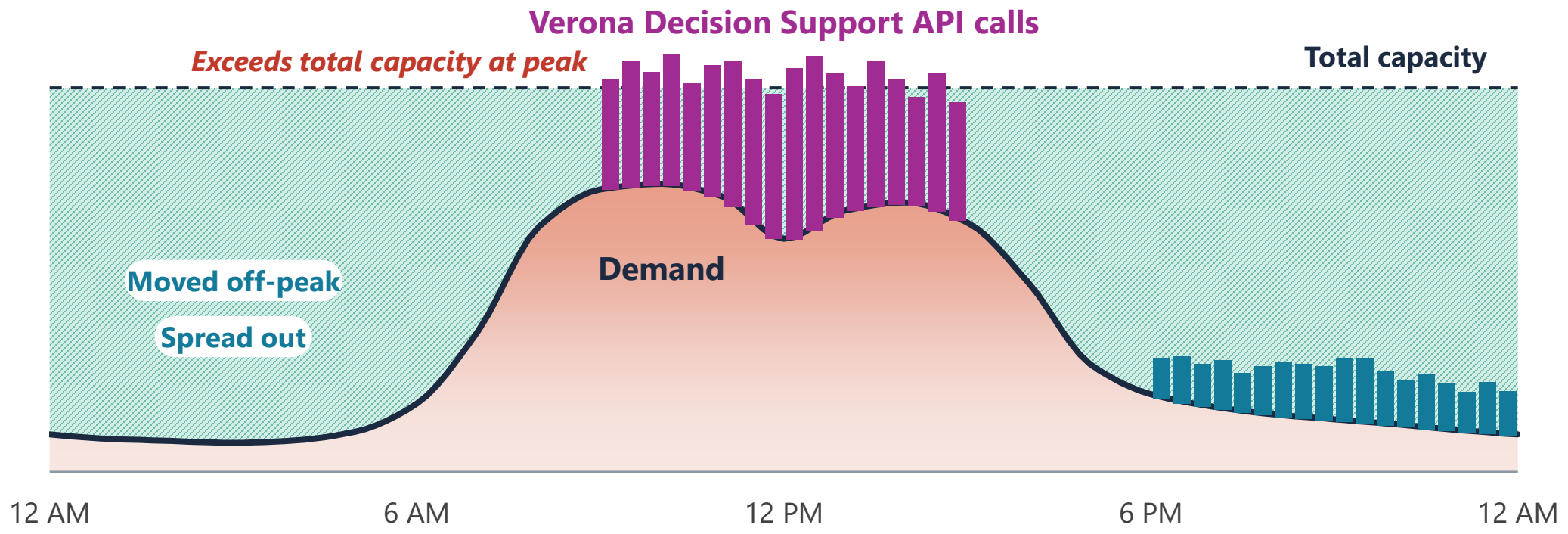
Overnight batch

- Problem list
- Immunizations
- Procedures
- Family history
- Social history
- Care plans

During the encounter

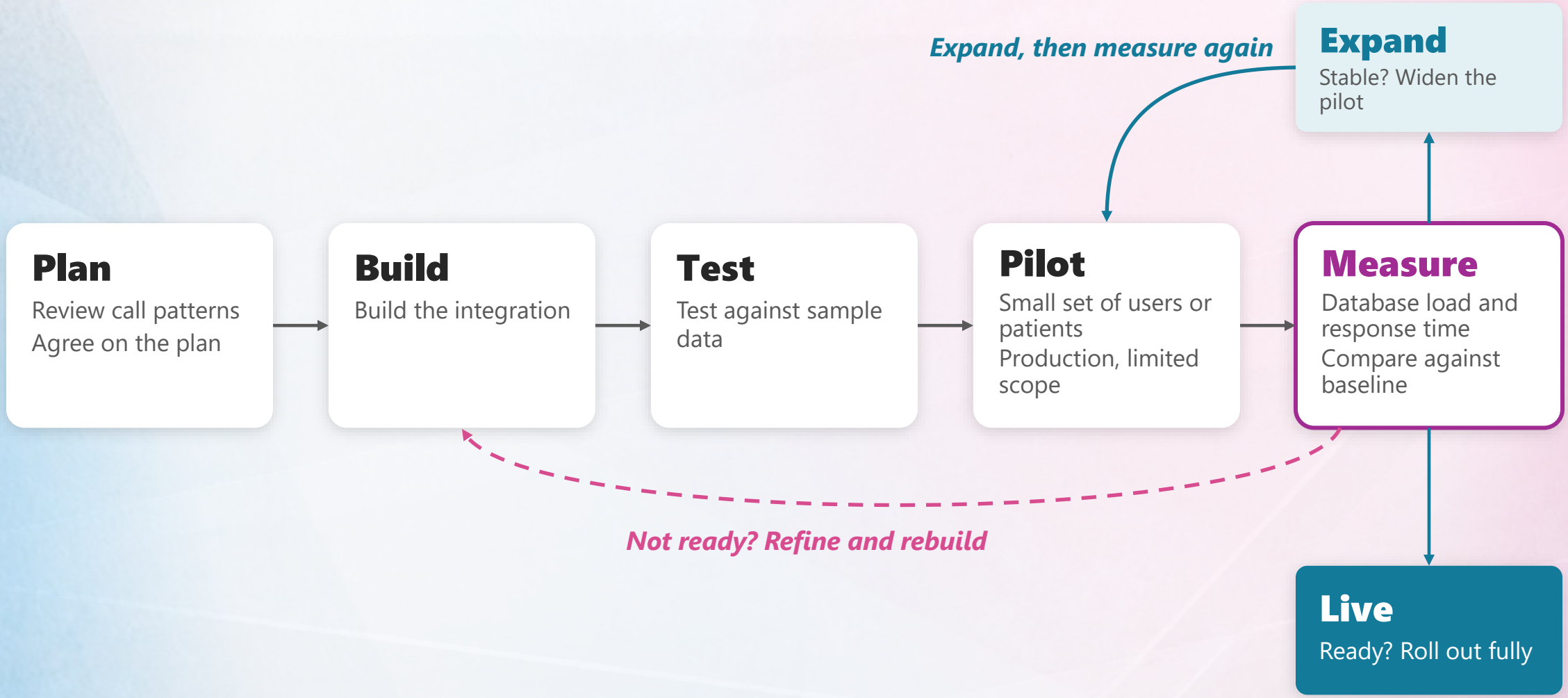
- Vital signs
- Active medications
- Allergies
- New orders

Peak Hours


Verona Decision Support API calls
Exceeds total capacity at peak
Total capacity
Moved off-peak
Spread out
Demand
12 AM
6 AM
12 PM
6 PM
12 AM

How to Implement Well

How Health Systems Implement Safely



What to Have Ready for Your Health System



Call patterns

Which endpoints you call, how often, and in what sequence.



Call volume

Requests per day at go-live and at full rollout.



Rollback plan

How the health system throttles or turns the app off.



Triggers

What user action or schedule starts each call.



Peak behavior

What your app does during the busiest hour of the day.



A name

Who from your team joins the pilot and reads the numbers.

Key Observations



Data exchange is growing



Scaling data exchange is hard



There are strategies to scale data exchange efficiently

Takeaways



Use the right tool



Exchange minimum necessary data



Avoid peak hours



Implementing is iterative



Be prepared to collaborate

Resources

fhir.epic.com

API documentation, specifications, and sandbox

fhir.epic.com

Designing an Efficient Application

Q&A

Q&A

OEC32

Wrap-Up: Putting It All Together

Form closes today at 5:00 PM

© 2026 Epic Systems Corporation. All rights reserved. PROPRIETARY INFORMATION - This item and its contents may not be accessed, used, modified, reproduced, performed, displayed, distributed or disclosed unless and only to the extent expressly authorized by an agreement with Epic. This item is a Commercial Item, as that term is defined at 48 C.F.R. Sec. 2.101. It contains trade secrets and commercial information that are confidential, privileged, and exempt from disclosure under the Freedom of Information Act and prohibited from disclosure under the Trade Secrets Act. After Visit Summary, ASAP, Aura, Beacon, Beaker, Beans, BedTime, Best Care Choices for My Patient, Bones, Break-the-Glass, Bugsy, Caboodle, Cadence, Canto, Care Everywhere, Charge Router, Cheers, Chronicles, Clarity, Cogito ergo sum, Cohort, Comfort, Community Connect, Compass Rose, Cosmos, Cosnome, Cupid, Discovery, Epic, EpicCare, EpicCare Link, Epicenter, EpicShare, EpicWeb, Epic Earth, Epic Nexus, Epic Research, Garden Plot, Grand Central, Haiku, Happy Together, Healthy Planet, Hello World, Hey Epic!, Hyperdrive, Hyperspace, Kaleidoscope, Kit, Limerick, Lucy, Lumens, MyChart, Nebula, OpTime, Phoenix, Powered by Epic, Prelude, Radar, Radiant, Resolute, Revenue Guardian, Rover, Share Everywhere, SmartForms, Sonnet, Stork, System Pulse, Tapestry, Trove, Welcome, Willow, Wisdom, With the Patient at Heart, and WorldWise are registered trademarks, trademarks, or service marks of Epic Systems Corporation in the United States of America and/or other countries. Other company, product, and service names referenced herein may be trademarks or service marks of their respective owners. Patents Notice: www.epic.com/patents.