A FIELD GUIDE FOR SCHEDULERS

Why is my P6 slow? What to check before anyone proposes a bigger server.

When P6 slows down, it is hard to know where to start. These twelve pages help schedulers find the cause, run practical checks and know when to bring in IT.

WRITTEN FOR

Schedulers, planners and project controls leads running Oracle Primavera P6 EPPM or Professional on more than a handful of projects.

FROM

LoadSpring, hosting and managing P6 environments since 1999. November 2026.

P6 slows down in four places. Find yours before you spend anything.

Every P6 click travels the same path: the client, the network, the job services and web tier (EPPM web, job services, Citrix; P6 Professional talks straight to the database), the database. Almost every performance complaint starts in one of those four places, and each one has a check that costs nothing. This guide takes them in the order that finds the problem fastest. Each check says whether you can run it yourself or should ask IT.

What is in here

The request path, and why it matters	3
Layouts: the quickest test to run	4
The F9 checklist	5
Imports, exports and the job services queue	6
Remote users and network delay	7
Slow by mid-afternoon	8
After the upgrade	9
Ten questions for your database administrator	10
When it is the environment	11
The one-page checklist	12

Three rules the guide follows

- **Free before paid.** Every chapter opens with a check that needs no purchase order. Most slow environments are fixed there.
- **Schedule quality and infrastructure are different problems.** A schedule with too much logic is slow on any server. Check which one you have before fixing either.
- **Measure from where the slow people sit.** The scheduler in the field and the database administrator in the data center are looking at different machines.

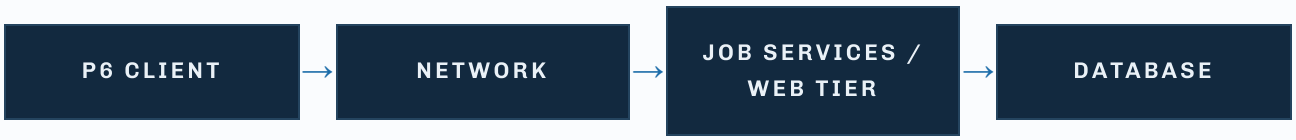
ABOUT THE AUTHOR

This guide is written by a company that hosts P6 for a living. Chapters 2 to 8 will fix most problems without us. Chapter 9 explains when a change of environment may help, and when a capable in-house team is well placed to keep managing it.

CHAPTER 1

Where P6 can slow down, and why "slow" is not one thing.

When someone says P6 is slow, they are describing a symptom that could originate in four different places. The same symptom, four different fixes, four different budgets.



Every screen refresh, every F9, every import travels this path. P6 Professional in particular is a chatty client: one action can be many round trips, so anything that slows one hop is multiplied.

SYMPTOM	USUALLY POINTS AT	CHAPTER
Large schedules open slowly	Database read performance, and how much the layout asks for	2, 8
Scheduling (F9) takes forever	Schedule size and logic first; server capacity second	3
Imports and exports time out	The job services tier, or the database absorbing writes	4
Remote users lag, office users are fine	Latency between client and database	5
Fine at 8am, crawling by 3pm	A server resource that fills up as the day runs	6
Slower since the last upgrade	Statistics and indexes not rebuilt after a schema change	7
Slow for everyone, all the time	An environment that may have outgrown what it was installed on	9

THE ONE RULE

Before anyone proposes a bigger server, find out which of the four places the problem is in. Fixing the cause beats paying for capacity: a faster server does nothing for a network delay, a heavy layout or stale database statistics, and all three survive an upgrade untouched.

CHAPTER 2

Layouts: the quickest test to run.

Opening a large project makes P6 pull the whole activity set, its relationships, its codes and the layout that renders them. On big projects the layout is often the multiplier, not the data.

You can check: the layout test

Open the same project twice: once with the layout people actually use, once with a minimal layout. Minimal means no group-and-sort on activity codes, columns cut to a handful, filters off, no resource or relationship columns. If the minimal layout is dramatically faster, the layout is the cost. If it is not, the layout is not the cause; move to chapter 8.

What makes a layout expensive

- **Group-and-sort on activity codes** across a large project, especially several levels deep.
- **Columns that compute:** anything summarized, rolled up or derived from relationships or resources.
- **Filters that scan** rather than index: text contains, complex OR chains.
- **Gantt with relationship lines on** for thousands of activities.
- **Layouts shared globally** that were built for one project and inherited by everyone.

Habits that keep layouts fast

- [ ] Publish two or three sanctioned layouts per role and retire the rest.
- [ ] Give heavy analysis layouts a name that says so, and a "lean" twin for daily work.
- [ ] Review global layouts quarterly; they accumulate.
- [ ] Train new schedulers on the lean layout first.

IF THE LEAN LAYOUT IS STILL SLOW

Then the layout is not the cause. Ask IT when database statistics were last gathered (chapter 7) and how the database was sized against today's activity counts (chapter 9).

CHAPTER 3

The F9 checklist.

Scheduling time grows with relationship count, constraint count and calendar complexity, not just the number of activities. A schedule that F9s slowly is often carrying thousands of redundant links or a calendar per contractor.

You can check: read the schedule log first

The log tells you what F9 actually processed. Look at the relationship and constraint counts against the activity count. A schedule with far more relationships than activities, or constraints on a large share of activities, is a schedule-quality problem, not an infrastructure problem.

- ☐ **Redundant logic.** Relationships that duplicate a path already implied by others. Remove them; the dates do not change, the F9 does.
- ☐ **Constraint sprawl.** Mandatory and start-on constraints on activities that should be driven by logic. Each one is an exception the engine must honor.
- ☐ **Calendars.** One calendar per contractor, per shift and per site multiplies working-time calculations. Consolidate to the few that differ materially.
- ☐ **Open ends.** Activities with no predecessor or successor. They do not slow F9 much, but they suggest a schedule nobody has audited.
- ☐ **Out-of-sequence progress.** Retained logic on hundreds of out-of-sequence activities is expensive; check the scheduling option and the count.
- ☐ **Multiple float paths and critical path options.** Turn off what the current analysis does not need.

RULE OF THUMB

If the log shows excess logic, fix the schedule. If the log is clean and F9 is still slow, the server doing the work may be the constraint, and that is chapter 9.

WHAT LOADSPRING SIZES FOR

Scheduling is one of the workloads a P6 environment is sized for specifically, against the real schedules it carries, not the seat count.

CHAPTER 4

Imports, exports and the job services queue.

XER and XML exchange runs through P6's job services on the application tier. When imports queue or time out, the job tier is short of capacity, the file is competing with other scheduled jobs, the database is slow to absorb the writes, or the network between the machine holding the file and the server doing the processing is slow or dropping packets.

Ask IT to check: the job services queue

Ask IT to look at the job services queue at the moment imports fail. If summarizer or scheduler jobs are stacked, your import is waiting in line, not failing. Then try the same file at a quiet hour. If it succeeds, capacity is the issue, not the file.

What shares the tier

- Project summarizer runs
- Scheduler and leveler jobs
- XER and XML imports and exports
- Reporting extracts and publications
- Anything scheduled "overnight" that has drifted into the working day

Spread before you scale

- [ ] Map every scheduled job to the hour it runs. Most environments find three things at 9am.
- [ ] Move summarizer runs out of the import window.
- [ ] Set import expectations: large XERs at quiet hours, not before the status meeting.
- [ ] If the queue is still congested after spreading, the tier may be undersized (chapter 9).

WHY THIS ONE IS UNDER-PROVISIONED SO OFTEN

Job tiers are invisible until they queue. They are rarely sized at install, and they are often the first thing a growing environment saturates. A well-run environment separates and sizes them deliberately.

CHAPTER 5

Remote users and network delay.

P6 Professional makes many round trips to the database for a single action, so every millisecond of network delay is multiplied. Office users on the local network rarely notice it. VPN, field and overseas users usually do.

Ask IT to check: response times from both places

Ask IT to compare response times (ping) to the database server from an office machine and from a remote one, at the time the remote user complains. If the remote delay is tens of milliseconds higher, a server upgrade will not fix the experience. The distance is the problem.

1	many	×
ACTION IN THE CLIENT	ROUND TRIPS TO THE DATABASE	NETWORK DELAY, EVERY TIME

The fixes that work

- **Run the client next to the database.** Publish the P6 client from the same data center as the database, so the round trips happen inside one building and the user receives screen updates only. This is how hosted P6 serves field teams.
- **Use the web client where the workflow allows.** P6 EPPM web pages make fewer round trips per action than Professional for many tasks.
- **Do not** buy a bigger database server for a network delay. It will be faster in the office and identical in the field.

FIELD EXAMPLE

Tutor Perini's international unit ran Primavera from sites eight time zones from headquarters on intermittent connections. A published, hosted client delivered in a week what a twelve-week infrastructure build was estimated to take.

CHAPTER 6

Slow by mid-afternoon.

Fast at 8am and crawling by 3pm usually means a server resource that fills up as the day runs: application server memory, temporary database space, or database connections set up for fewer simultaneous users than actually arrive.

You can check: when it slows and what else runs then

Note the time it degrades and ask IT what else runs then. Summarizer jobs, reporting extracts and backups scheduled into working hours are the usual suspects. If restarting services fixes it until tomorrow, the environment is probably short of a resource for the day's load.

Resources that fill

- **Application server memory and sessions** that are not released as users come and go.
- **Temporary database space** consumed by long-running reports and summarizer runs.
- **Database connections** configured at install for the user count of the day.
- **Log and archive space** that nobody rotates.

What to record for a week

- ☐ Concurrent user count by hour.
- ☐ Time of first complaint each day.
- ☐ Jobs and backups running in the hour before it.
- ☐ Whether a service restart resets it.

WHY THIS NEEDS MONITORING, NOT GUESSING

Sizing for simultaneous users needs measurement over time, which needs monitoring that many environments do not have. Continuous performance monitoring is standard on LoadSpring Managed Hosting, so the pattern is seen before users report it.

CHAPTER 7

After the upgrade.

P6 upgrades change the database structure. If statistics and indexes are not rebuilt afterwards, the database plans its queries against a map of tables that no longer exists, and everything gets slower even though nothing looks broken.

Ask IT to check: the post-upgrade database tasks

Ask whether statistics were regathered and indexes rebuilt after the upgrade, and whether the vendor's post-upgrade database tasks were run to completion rather than abandoned when the upgrade window closed. This is the most common cause of environment-wide slowness after a version change, and it is free to fix.

- [ ] Statistics gathered on the P6 schema after the upgrade, not before it.
- [ ] Indexes rebuilt or validated; fragmented indexes identified.
- [ ] Vendor post-upgrade scripts run to completion, with output kept.
- [ ] Job services, summarizer and publication settings re-checked; upgrades reset some of them.
- [ ] A before-and-after timing of the three slowest common actions, so the conversation has numbers.

Rehearse the next one

Upgrade rehearsal against a clone, with performance compared before and after, is how this is caught in advance rather than discovered by the schedulers on Monday. Teams under time pressure often skip the rehearsal; on LoadSpring Managed Hosting it is part of the routine.

ALSO AFTER HEAVY DATA LOADS

Statistics go stale after large imports and end-of-period consolidations too, not only after upgrades. If the environment slows after a big migration or a merger of databases, regather first.

CHAPTER 8

Ten questions for your database administrator.

These help your database administrator investigate performance and explain what is being monitored. You do not need to know the answers yourself. Ask them in order; each one closes a chapter of this guide.

1. **When were statistics last gathered on the P6 schema,** and is it scheduled or manual?
2. **Were the vendor's post-upgrade database tasks run to completion** after the last version change? Is the output kept?
3. **How many activities does the environment carry today,** and what was it sized for at install?
4. **What is the response time to the database** from the office, from the VPN, and from the site that complains most?
5. **What runs on the job services tier, and when?** Can we see the queue at 9am on a Monday?
6. **How are database connections and application server memory sized** against measured peak usage, not license count?
7. **What is backed up into working hours,** and what would it take to move it?
8. **Is there performance monitoring,** and can it show me last Tuesday at 3pm?
9. **Where would we rehearse the next upgrade,** and how long would it take to stand that environment up?
10. **If the database host failed this afternoon,** what is the tested recovery time and the data loss window?

IF THE ANSWERS ARE QUICK AND SPECIFIC

You have a well-run environment and a real capacity question. Chapter 9 applies.

IF THE ANSWERS ARE "WE WOULD HAVE TO CHECK"

The environment is being looked after reactively. That is not a criticism of the team; it is a sign of how many hours the environment demands, which chapter 9 puts a number on.

CHAPTER 9

When it is the environment.

If statistics are fresh, layouts are lean, the queue is clear and network delay is low, and P6 is still slow for everyone all the time, the environment itself may have grown past the infrastructure it was installed on, usually years ago, one project at a time.

Ask IT to check: today's volume against the install sizing

How many activities does the environment carry now, versus what it was sized for at install? Many teams find the install assumptions were passed long ago and nobody re-sized, because re-sizing is a project in itself.

When a change of environment may help

A change of hosting environment is worth considering when P6 is slow despite clean checks, when the hours of internal upkeep are being spent anyway, and when the next upgrade has no rehearsal environment. Sizing against the schedules you run rather than seat count, separating and sizing the job tier, publishing the client next to the database, and monitoring usage continuously are the work LoadSpring Managed Hosting takes on. Your configurations, layouts and integrations come with you: this is not a move to a different product.

WHEN IT IS NOT

A capable in-house team with a stable environment and clean answers to chapter 8 is well placed to keep managing it. We say so on our comparison page and we mean it here.

Work down, stop at the first thing that fixes it.

- [ ] Open the project with a minimal layout. Faster? Fix the layouts.
- [ ] Read the schedule log. Too many relationships and constraints? Fix the schedule logic.
- [ ] Check the job services queue when imports fail. Stacked? Spread the jobs.
- [ ] Ask IT to compare response times from the office and from remote sites. Much higher remotely? Publish the client next to the database.
- [ ] Note when it degrades daily and what runs then. A restart fixes it? A resource is filling up.
- [ ] Statistics and indexes rebuilt after the last upgrade? If not, do it now.
- [ ] Ask the ten questions. Quick answers point to a capacity question; slow answers point to an hours question.
- [ ] Still slow with everything clean? Count activities against the install sizing. The environment may have outgrown it.

Two free tools that put numbers on it

The P6 diagnostic at loadspring.com/tools/why-is-p6-slow: pick the symptoms you see and it shows where they usually point. **The TCO calculator** at loadspring.com/tco-calculator prices the hours of internal work for your own environment.

If you want it looked at

Talk to a P6 expert at loadspring.com/talk-to-an-expert: a conversation about your environment, your requirements and your timeline, with a plan and a quote to follow.

ABOUT LOADSPRING

Managed Hosting for project applications since 1999, Primavera P6 longest of all. Headquartered in Portsmouth, New Hampshire. SOC 2 Type II and Cyber Essentials Plus certified, GDPR compliant, hosted in ISO 27001 certified data centers. Twenty-four published customer stories at loadspring.com/customers.