

CHEAT SHEET

Best practices in AI procurement



Red flags in legacy contracts

If you see these, you're trapped:

- Long-term (5+ years) with no off-ramp or no/costly exit clause
- Vendor owns your data and/or custom code
- Payment tied to deliverables, not outcomes
- Auto-renewal windows buried in fine print
- Change requests are costly/require 90+ day amendment process
- No performance benchmarks or uptime guarantees
- One-time training with no provision for staff turnover



What good contracts look like

Instead of this...



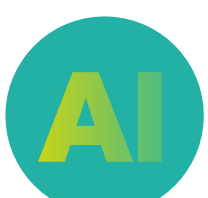
Do this...

- | | |
|-------------------------|--|
| • 60-month fixed term | • Short initial term + outcomes/performance-based renewals |
| • Opaque usage fees | • Transparent & flexible pricing structure |
| • Vendor owns data | • State owns all data and custom code |
| • Pay for deliverables | • Pay for measurable outcomes |
| • Lock specs at signing | • Build in iterative review cycles |
| • One-time training | • Ongoing training tied to need |
| • Proprietary system | • Open APIs, open standards, integration-ready |
| • Exit penalty | • 60-90 day termination clause |



Writing better solicitations

- Lead with the problem, not the fashionable technology. Describe what is broken. Let vendors propose solutions.
- Buy outcomes, not features. "Reduce documentation time by 40%" is better than "must have voice-to-text transcription." Know what good outcomes & feasible KPIs look like.
- Stay vendor-neutral. Avoid naming specific platforms, models, or tech stacks in requirements.
- Require a pilot. A paid proof-of-concept period before full deployment protects the state and tests vendor claims.
- Make sure to account for ongoing support and use costs. Budget for maintenance and model updates.
- Build your evaluation criteria before you read proposals and weight them explicitly.



Specific to AI procurement

- Ask vendors: how are you sourcing, managing, and governing the data? Who owns data used to train the AI model?
- Ask for real-world data, not controlled-environment benchmarks.
- Require transparency in how the AI makes decisions, especially in high-stakes use cases.
- Check for bias, especially if AI touches eligibility or prioritization decisions
- Insist on human oversight. AI should assist workers, not replace their judgment.
- Confirm the system works for your actual users: staff, field workers, citizens



Three questions to ask every vendor

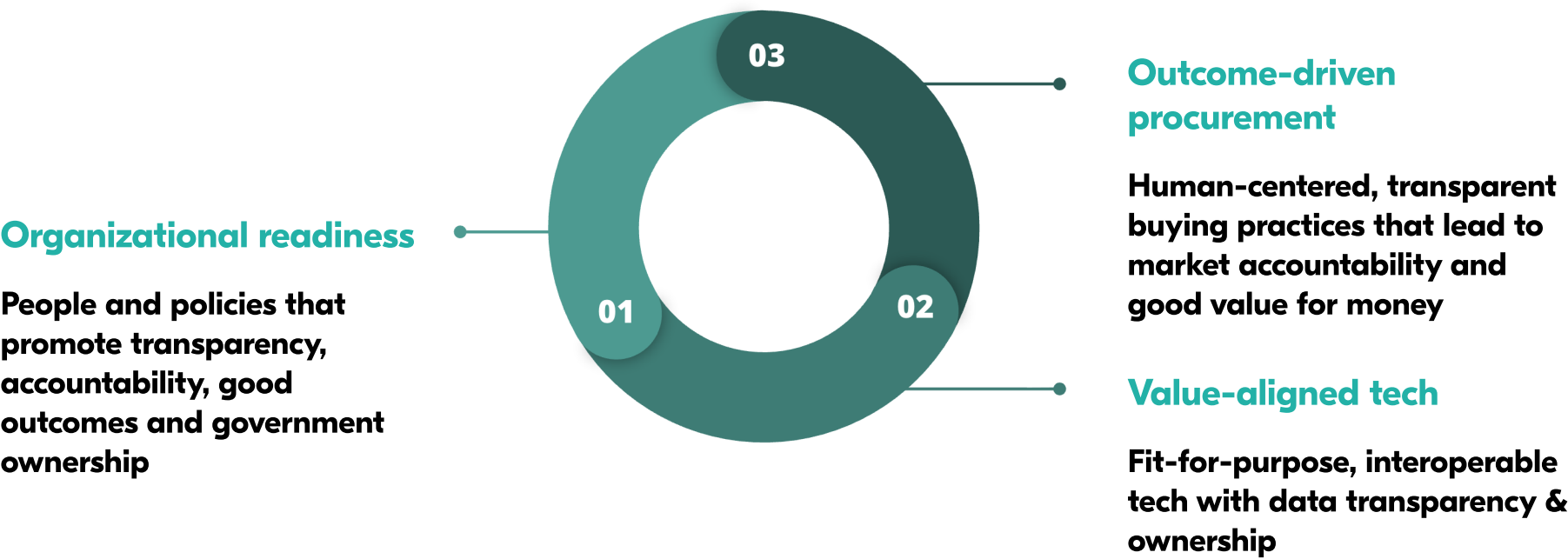
1. "Who owns the data, and can we take it with us if we leave?"
2. "Can you show us this working in a state agency like ours, not a demo?"
3. "How do you address model improvements, updates, and what do they cost?"

Bonus question if receiving a free pilot: "What is your usual fee structure?"



PROCUREMENT PRACTICES

To get the best from AI, you need...



Planning

- **Understand the problem** you are trying to solve, and what solutions are out there. AI might not be the best fit! Be clear on your use case and what good looks like.
- **Engage your key stakeholders.** This includes internal users, your residents, suppliers, and the public. Develop use cases with them!
- **Think through your budget.** Don't just consider up-front costs. Preparing data, maintaining the tool, and moving from pilot to scale all require resources. Consider government-wide not with a per-seat enterprise contract.



Solicitation

- Involve your procurement, technology, and department staff early and often!
- Ask the market what is possible and don't fall in love. RFIs, market research, past performance all matter. Take advantage of agile vendor pools, NASPO or GSA contracts.
- Supporting testing and innovation. Encourage pilots and sandbox environments for safe learning.
- Use enterprise licenses as your foundation, not your final destination: Consider if your project/service teams can build custom prompts, models, or applications using the enterprise-license.



Evaluation

- Look at process for evaluating AND technical criteria.
- Process: focus on fast & rigorous
 - Bake offs, sandboxes, show & tell, always check references with other governments
 - Make sure you have the right people in the room.
 - Technical: focus on interoperability & implementation realism
 - API-first interoperability
 - Modularity
 - Business rules engine configurable by state staff
 - Low-code policy adjustments
 - Clear integration architecture



Contract management

- **Pay for outcomes:** Tie payments to tested functionality and go-live results.
- **Enforce SLAs:** Set clear uptime, defect, and response targets with penalties.
- **Time-box delivery:** Fixed checkpoints with escalation if milestones slip.
- **Independent oversight:** Reviews to validate quality and compliance.
- **Protect exit rights:** Require open APIs, full data access, and transition support.
- **Strong governance:** Weekly delivery reviews + executive steering with decision authority.

