

The Future Of TuringBots

TuringBots Will Be Development Teams' Best Companions: Planful, Generative, And Insightful

July 21, 2023

By Diego Lo Giudice with Chris Gardner, Mike Gualtieri, Brian Hopkins, Carlos Casanova, Christian Prandi, Merve Kandemir, Kara Hartig

FORRESTER®

Summary

Generative AI TuringBots speed and improve software development. But the siren song of generative AI to build and deploy applications in seconds challenges tech executives to identify realistic opportunities. Short-term, TuringBots will assist all dev roles. Midterm, TuringBots will become better peers to humans and more accurate. Long-term, they will become more autonomous with additional reasoning capabilities and human supervision (still), but productivity will reach unimaginable heights. This report examines the timeline for major TuringBot developments, helping technology executives plan an adoption roadmap and align practices, governance, and benefits.

TuringBots Boost Your Development Teams' Productivity

Software development, despite past and current automation efforts, is still largely manual. Agile and DevOps in the past few years have enabled faster deployments. According to [Forrester's Developer Survey, 2022](#), 31% of developers report they release large or entirely new applications through automation quarterly, and 15%, monthly — but with no real breakthroughs. On the other hand, expensive professional developers can write on average no more than 10 lines of [code](#) daily due to a multitude of technologies, cognitive overload, suboptimal development processes, countless meetings, reviews, email, and more. Technology execs are under pressure to deliver more better, faster, and possibly cheaper. That is all about to change with [TuringBots](#), with early adopters claiming increased software development productivity of 15% to 30%. Forrester defines TuringBots as:

AI-powered software that augments application development and infrastructure and operations (I&O) teams' automation and semiautonomous capabilities to plan, analyze, design, code, test, deliver, and deploy while providing assistive intelligence on code, development processes, and applications.

Assistive bots outside of the software development lifecycle (SDLC) are not what Forrester calls TuringBots. TuringBots are specialized bot assistants specifically created for development teams and their stakeholders. Microsoft has released Copilots for non-software-development-related roles and activities, but they are not TuringBots; they are assistive bots for all business and technical users across the enterprise. Google Bard and OpenAI ChatGPT are both business and development assistive bots.

Generative AI Advances TuringBot Maturity Much Faster Than Expected

AI has progressed at the speed of light in the past six to 12 months: from single models to multimodels with large language and foundational models that effectively generate assets like text, images, and code for the first time in the history of AI. That progress can be summarized in two words: [generative AI](#) (genAI). In 2020, when Forrester first coined the term "[TuringBots](#)," we predicted a long journey of five to 10 years before seeing a concrete impact in the market. In 2023, we've shortened that journey to two to four years. We are already seeing value and benefits, and TuringBots are improving at a quick pace. Thanks to genAI, with TuringBots, business and technology management together can:

- **Boost speed thanks to improved natural language processing.** Product development teams have learned to interact with computers in an unnatural way

Not Licensed For Distribution.

© 2023 Forrester Research, Inc. All trademarks are property of their respective owners.

For more information, see the [Citation Policy](#), contact citations@forrester.com, or call +1 866-367-7378.

with a mouse, windows, widgets, clicks, and command lines to get work done. TuringBots like ChatGPT offer development capabilities through straight natural language (NL) statements called “prompts.” Modern integrated development environments (IDEs) are getting genAI integrations so that developers can use NL prompts or comments to ask the TuringBot for code suggestions, as GitHub Copilot X does in Microsoft Visual Code. Project collaboration tools like Atlassian and testing tools like Parasoft and Quinnox are building plug-ins to ChatGPT to allow team members to interact with their capabilities in NL. All software development lifecycle (SDLC) stakeholders sooner or later will leverage an invaluable TuringBot assistant with genAI to accelerate work.

- **Bring knowledge to developers faster than ever.** Before writing code, developers spend more time finding the right information, API, signature, and code libraries than writing the code itself. TuringBots deliver crucial knowledge and information to developers at light speed. Search engines integrated with genAI make finding the information much easier — no more reading thick manuals or navigating websites. A TuringBot like GitHub Copilot, Tabnine, or Amazon CodeWhisperer plugged into the IDE provides the information through a simple NL prompt or comment with code suggestions for the request.
- **Generate all sorts of product development assets in seconds.** Building applications and software products requires the design and development of many software assets. Most of the build process for assets is manual or at best semimanual. To build an application, for example, you have to design the user experience, interface layout, and APIs to invoke. You also have to design test cases and build scripts to test the application, data structures, and documentation. If you are in a cloud-native platform, perhaps you even have to build some infrastructure-gluing code. TuringBots not only can potentially generate or at least stub many artifacts in a fraction of a second from NL prompts but also generate code directly from a UI sketch as GPT-4 does. IBM/Red Hat Project Wisdom on OpenShift and Google Duet AI on Google Cloud Platform can also generate design artifacts (e.g., an API description) for infrastructure and software pipelines.

TuringBots Accelerate All Aspects Of The SDLC, Not Just Coding

TuringBots are appearing across the entire SDLC and in [Forrester’s Modern Application Development Model](#). The spread of TuringBots across all development steps gives technology executives the opportunity to leverage a newer, faster, and even more agile development journey. TuringBots can collaborate with all development team members (and other business and IT stakeholders), reducing friction, connecting different stages

Not Licensed For Distribution.

© 2023 Forrester Research, Inc. All trademarks are property of their respective owners.
For more information, see the [Citation Policy](#), contact citations@forrester.com, or call +1 866-367-7378.

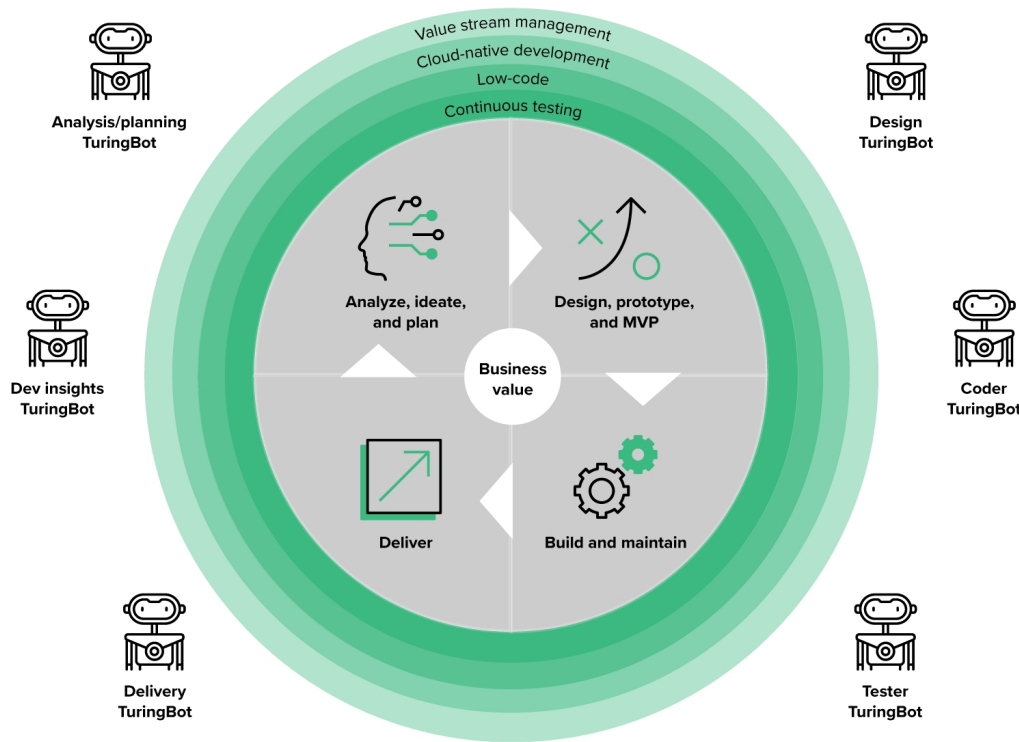
of the lifecycle, and short-circuiting some to build applications and products faster. They will do so by automating many use cases with high and differentiated productivity gains (see Figures 1 and 2). Low-code, cloud-native development, and value stream management vendors have all been working on AI and are now infusing or have plans to infuse genAI into their products. At each step of the lifecycle, TuringBots can help:

- **Analyze and plan software projects.** TuringBots can help product managers and product owners plan and assess development teamwork, predict the ability to meet delivery deadlines, optimize and reprioritize requirements based on goals, generate documentation, and more. Intelligence is Atlassian's TuringBot that leverages a graph of metadata and that Atlassian has developed by working over the years with hundreds of thousands of clients. Now in partnership with OpenAI, Atlassian has integrated ChatGPT to improve teamwork collaboration assistance. Forrester calls this example an "analysis/planning TuringBot." These TuringBots are at an early stage of development but have high potential and will evolve very fast.
- **Automate design and models and generate code.** Tools like Figma are popular tools that business teams use to generate UI mock-ups and prototypes for web pages and apps and mobile UIs. Tools that generate HTML 5.0 or JSON code from the UI are not new, but generating a full web application from a handcrafted UI design is a breakthrough that only a genAI TuringBot like OpenAI GPT-4 or Microsoft Sketch2Code can do. TuringBots can also automate the deployment of [AI/ML models](#), making it easier for data scientists to build and deploy them. Forrester calls these types of TuringBots (e.g., DataRobot) "design TuringBots." The second largest ecosystem with high impact on the developer experience is code generation, and Forrester calls these AI-enabled tools "coder TuringBots." Examples are Replit Ghostwriter, GitHub Copilot, Tabnine, IBM Watson Coding Assistant, Amazon CodeWhisperer, Diffblue Cover, Google Bard and Duet AI, OpenAI ChatGPT and GPT-4, and Sourcegraph.
- **Make testing smarter and faster.** Software testing has become key in the SDLC ever since agile-plus-DevOps has been around. Why? Because the need to move testing from manual to highly automated is a milestone for continuous software delivery. But TuringBots can help automate and test smarter instead of hyperautomating all tests — increasing quality by automating more with less effort. This is the largest ecosystem of both traditional and new vendors in the TuringBots market. Forrester calls these tools "tester TuringBots." The ecosystem includes, for example, traditional testing companies like Tricentis, OpenText, Keysight Eggplant, Parasoft, Sauce Labs, UiPath, as well as AI-born testing

companies like Appvance, Functionize, Mabl, Tricentis Testim, and Quinnox Qyrus.

- **Accelerate and automate DevSecOps delivery.** Development, security, and operations (DevSecOps) has pushed professionals in these areas to become better friends, uniting them in a more efficient integrated process. The goal is to automate software pipelines that can deliver and deploy application features and code quickly and securely. Software pipelines are the enablement mechanism by which automation happens. “Deliver TuringBots” support developers and operations professionals in building those pipelines, applying DevSecOps principles, building infrastructure-as-code glue code, creating playbooks that meet business service-level agreements, and improving cloud development experiences. They can also help detect abnormal operating patterns to identify operational issues before they impact customers. The use cases are many. Examples of deliver TuringBots are Amazon DevOps Guru, Red Hat Ansible Lightspeed, Google Duet, GitLab UnReview to suggest code reviewers, Veracode, and Sync for security. Deliver TuringBots are a less mature product area but have high growth potential.
- **Drive product development with data insights.** Tools in software delivery pipelines generate huge amounts of data as developers go through the SDLC. “Dev insights TuringBots” can support a teams’ development strategy and process in real time based on that data and automate actions based on data in production (e.g., alerts, monitoring info). Dev insights TuringBots also help make better decisions based on real factual data instead of human intuition. They can also correlate better business value metrics to process improvements to help prioritize and optimize development activities, fine-tune specific work in the various steps of the lifecycle, observe code behavior in production, and suggest action/remediation based on real-time operational data. A dev insights TuringBot can even act as a “meta TuringBot” through which other TuringBots share and receive crucial data insights to optimize and further automate quality and security automation. This TuringBot might also offer planning capabilities like the analysis/planning TuringBot. Dev insights TuringBot examples are [CloudFabrix](#) and Jellyfish. The dev insights TuringBot could become the artificial intelligence for IT operations (AIOps) of the future, leveraging genAI. Today it’s still in its infancy.

Figure 1
TuringBots Augment All Role Capabilities To Build Better Products Faster



Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

Figure 2
TuringBots Reduce Time To Value In All Stages Of The Development Lifecycle

| | Use case examples | Key AI technologies/examples |
|------------|---|--|
| TuringBots | <ul style="list-style-type: none">• Improve user experiences• Create innovative designs and learn from experience• Optimize product/project plans• Modernize legacy underlying technologies• Improve cloud developer experience• Reduce time to value; do more with less | <ul style="list-style-type: none">• Analytics• ML• GenAI• Graphs• Examples: Amazon, Google, OpenAI, Microsoft, IBM |

Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

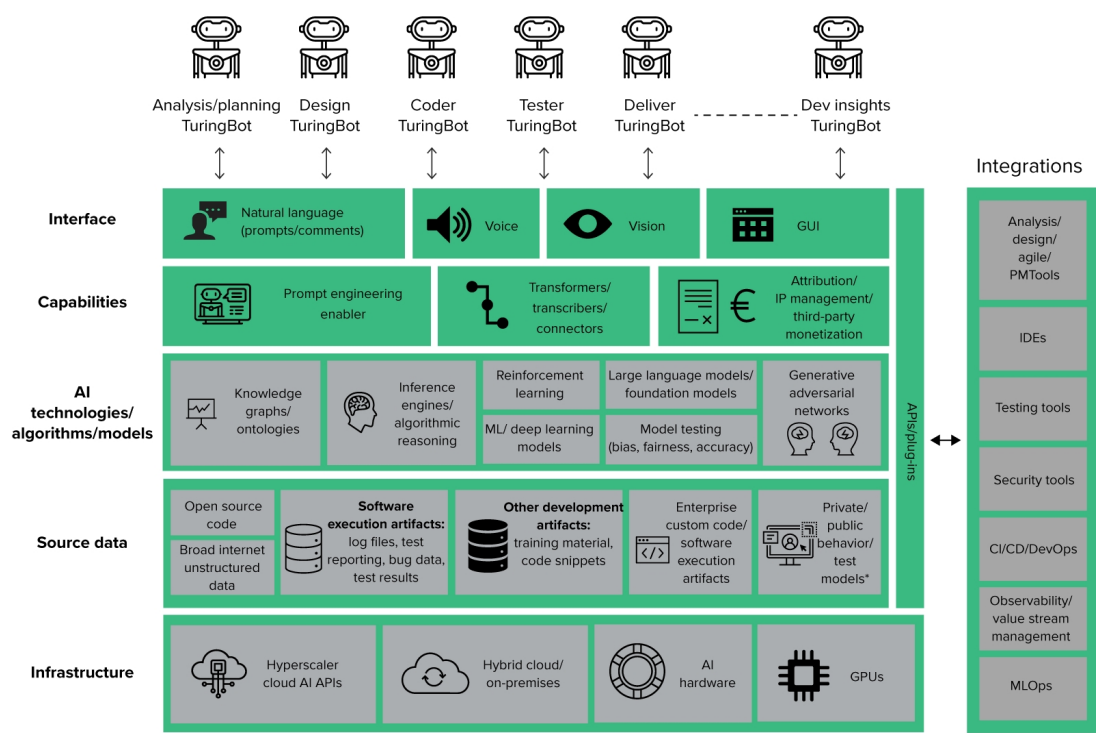
| | | | Key AI technologies/examples | | | |
|---|------------------------------------|---|--|-------|--------|----|
| | Type of TuringBot | Use case examples | Analytics | GenAI | Graphs | ML |
| → | Analysis/ planning TuringBot | <ul style="list-style-type: none">• Prioritize use cases to increase business value• Suggest product team composition• Predict project efforts/costs/ability to meet deadlines• Generate/optimize prompts and product specifications | | ✓ | ✓ | ✓ |
| | | | Examples: Atlassian Intelligence, OpenAI ChatGPT/GPT-4, Microsoft Copilot | | | |
| → | Design TuringBot | <ul style="list-style-type: none">• Generate UI mock-up from UI design artifacts• Generate web apps from UI design artifacts• Generate API designs/code from API descriptions | | ✓ | | |
| | | | Examples: OpenAI GPT-4, Microsoft Sketch2Code | | | |
| → | Coder TuringBot | <ul style="list-style-type: none">• Generate code in various programming languages• Generate code comments and documentation• Explain, teach, and suggest code solutions• Generate unit tests | | ✓ | | |
| | | | Examples: OpenAI ChatGPT/GPT-4, Bard, GitHub Copilot, Tabnine, Amazon CodeWhisperer, IBM Watson Code Assistant, Diffblue Cover, Google Duet AI | | | |
| → | Tester TuringBot | <ul style="list-style-type: none">• Improve product business/technical quality• Perform accessibility/usability testing• Increase business or technical testing coverage• Create functional tests, automation, and plans | | ✓ | | ✓ |
| | | | Examples: Tricentis Testim, OpenText Visual Testing, Appltools Eyes, Appvance, Parasoft, Testlio, Keysight Eggplant | | | |
| → | Deliver TuringBot | <ul style="list-style-type: none">• Strong engineering principles/guidelines• Increase delivery automations• Monitor/predict infrastructure resource requirements | | ✓ | | ✓ |
| | | | Examples: Amazon DevOps Guru, Red Hat Ansible Lightspeed, Google Duet AI | | | |
| → | Dev insights TuringBot | <ul style="list-style-type: none">• Predict ability to meet deadlines/suggest solutions• Predict bottlenecks/suggest solutions• Predict faults and suggest action/remediation• Generate business-value project progress reports | ✓ | ✓ | | ✓ |
| | | | Examples: CloudFabrix, OpenText ValueEdge | | | |

Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

Putting TuringBots To Work

Different bot types use different components of an overall TuringBots architecture — for example, the type of infrastructure that a TuringBot runs on (e.g., AI-specialized or unspecialized hardware, on cloud or on-premises) (see Figure 3). Furthermore, TuringBots require integration in development, testing, and delivery tools to be useful. It’s why OpenAI released AutoGPT.

Figure 3
The Anatomies Of TuringBots Differ Like Their Main Use Cases Do



*These include visuals, graphics, descriptive languages, DSL, etc.

Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

The Full Potential Of TuringBots Will Unfold Over Three Time Horizons

TuringBots will deliver impactful enterprise business ROI over three time horizons, with key factors influencing each:

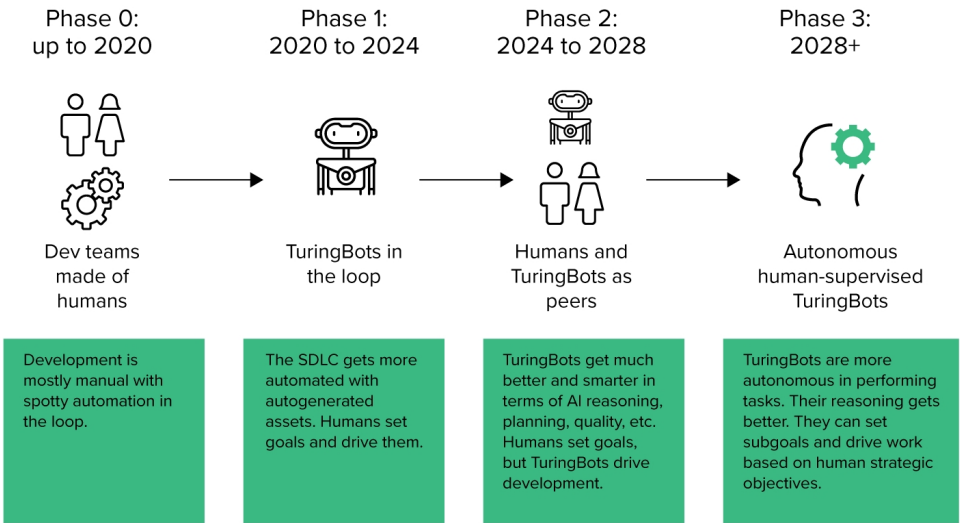
- **AI technology innovation speed.** The speed at which existing AI technologies improve and new ones emerge will impact ROI timing. We expect that mature TuringBots will require significant advances in algorithmic reasoning and planning beyond today's capabilities. However, just as the past six months have proven, disruptive technologies like genAI can be right around the corner and push massive change much faster than expected. A lot of innovation is happening around democratizing large learning models (LLMs) with open source and simplifying their use.
- **Enterprise change, governance, and practices.** What matters most is how the technology gets used once available. Accelerated adoption and scale often slow from a lack of enterprise risk management practices, governance, and cultural acceptance. A new SDLC is needed to effectively leverage TuringBots, but nobody knows exactly what it looks like. In addition, scaling TuringBots at enterprise development, not just developer or team, levels is still uncertain.
- **Creation of AI regulations, laws, and acts.** The creation of statewide regulations and acts at national and international levels could slow this emerging tech trend. AI and TuringBots are powerful technologies, and we need to make sure they are used safely for good purposes. This factor carries the most uncertainty for a short-term resolution.
- **Lack of trust in IP protection and security.** Clients in pharmaceuticals, financial services, manufacturing, and other industries will refrain from using any technology, including TuringBots, that uploads their differentiating and competitive crown-jewel traditional software. A global manufacturing CIO told us that even if vendors agreed to it in their product-license indemnification clauses, their lawyers would not allow the use of genAI coding tools right now. It took 10 years to convince banks to put data in the cloud, so how long will it take them to adopt at-scale TuringBots? Improvements in secure and high-quality code generation will influence adoption and increase trust too.

The Target State: From TuringBots In The Loop To Supervised Autonomous TuringBots

A precise estimation of when TuringBots will fully mature, take on 80% or more of coding and testing work, produce a large impact on business ROI, and increase productivity for the rest of the SDLC is impossible to determine. However, we have hypothesized what the target state of TuringBots will look like and how their journey will unfold over the next five-plus years, with ROI benefits that no one should ignore.

We believe TuringBots and humans will pass through major phases of collaboration maturity, which the fast pace of AI technology innovation will fuel. GenAI helped the TuringBot market take off to its current state. As AI technologies mature, TuringBots will traverse three key phases. In the first phase, TuringBots enter the loop with development teams' human stakeholders like product managers who remain in the driver's seat. In the next phase, TuringBots become members of the team and have more interactive collaboration with humans. TuringBots in the next (not necessarily the last) phase become more autonomous but still have human supervision (see Figure 4).

Figure 4
TuringBots Will Evolve In Three Phases From Servants To Peers And Autonomous Tools



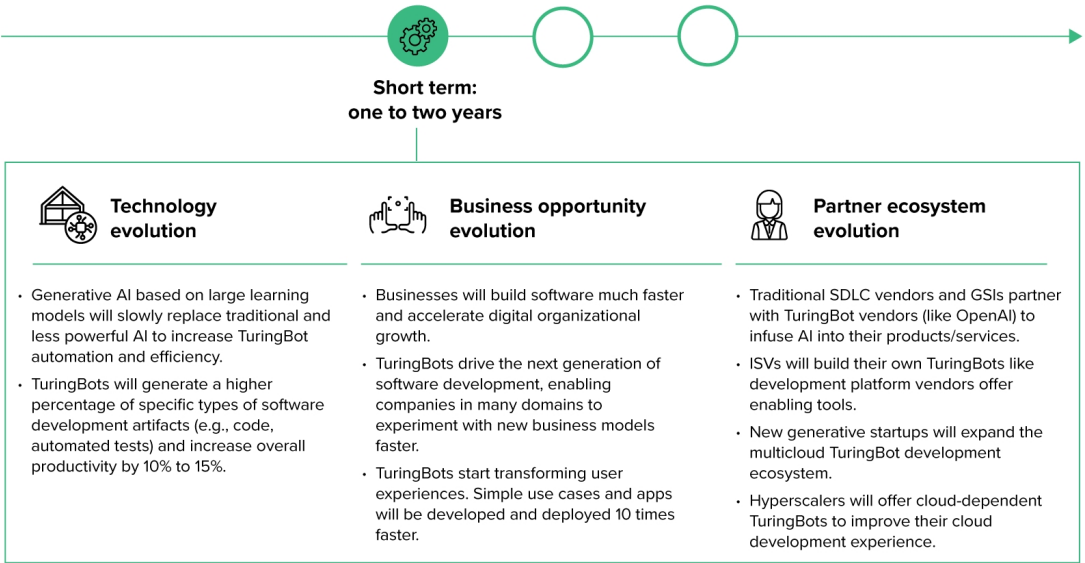
Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

The Short Term: TuringBots Enter The Loop With Low Productivity And Maturity

TuringBots in the short term (one year to two years) will get better at generating secure and high-quality software-related assets in the existing SDLC. Product owners, testers, and developers will leverage TuringBots as assistants, ordering them to accomplish tasks. Use cases might test the experience of an early UI design, predict outcomes of a project plan, write a piece of code, or write a test case. TuringBots in the SDLC will have varying levels of maturity and accuracy, productivity, and adoption among the various stakeholders. All assets that TuringBots generate will still need strict reviews and controls (see Figure 5).

We also expect that experiments with product implementations of TuringBots will be sparse across the lifecycle. Coding and testing will be the first stages of the SDLC to significantly increase productivity with coder and tester TuringBots. Our initial analysis of qualitative research on real use cases uncovered that testing productivity could improve by 15% to 30%; developers' coding productivity, by 30% to 50%; and overall SDLC productivity, by 10% to 15%. Businesses will experiment with, deploy, and change new applications in weeks and months.

Figure 5
The Short Term: TuringBots Will Be Siloed With Different Maturity And Adoption Levels



Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

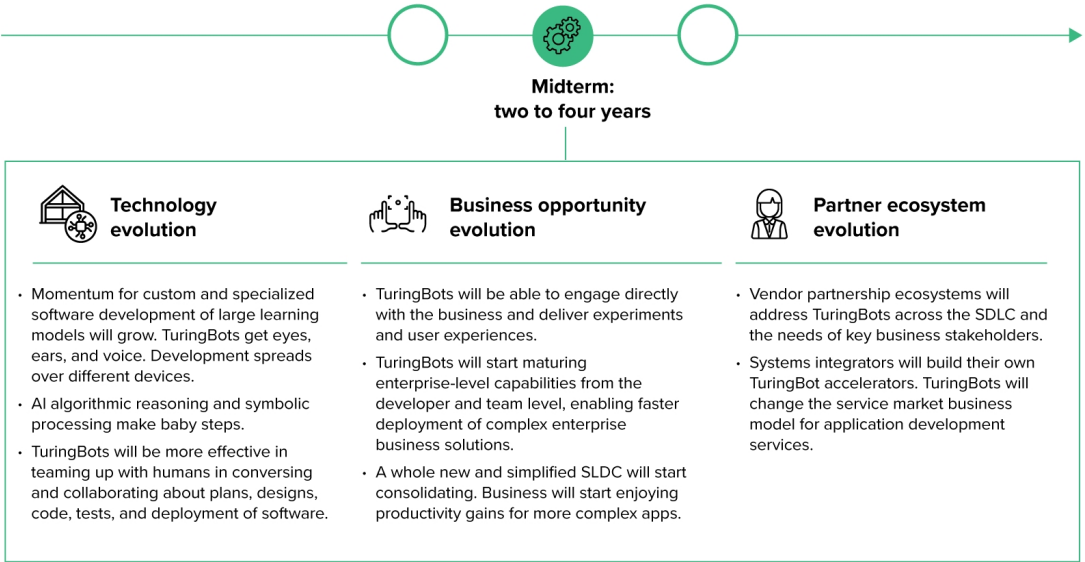
The Midterm: TuringBots Get Better, Become Peers To More Roles, And Boost Productivity

In the midterm (two to four years), all TuringBots categories will get better at what they do. They will leverage AI improvements like easy customization of LLMs and begin to engage in algorithmic reasoning and planning. TuringBots will get better at interacting with and suggesting next steps or corrective actions to product managers and owners, application and infrastructure coders and testers, and I&O professionals. Their level of maturity will reach the point of autonomously supporting businesspeople. As AI regulations get approved, TuringBots will start to comply. Human developers will increasingly trust TuringBots to support business stakeholders (see Figure 6).

TuringBots will also increase intelligence across the lifecycle. Analysis/planning TuringBots will improve early cycles of product development, increasing average testing productivity by 30% to 50% (sometimes even much more than 50%) and coding productivity by 50% to 80% (100% of plumbing code). Deliver and dev insights TuringBots will start addressing cross-role collaboration work, increasing overall product development productivity on average by 30% to 50%. Business will experiment

with, deploy, and change new solutions in days and weeks.

Figure 6
The Midterm: TuringBots Mature, Get Smarter, And Team Up With More Stakeholders



Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

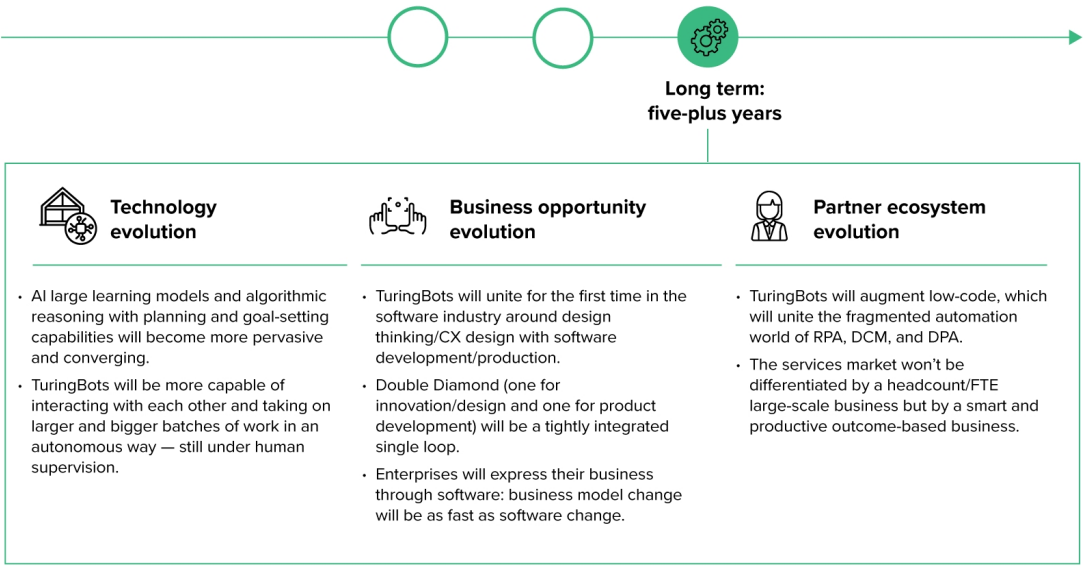
The Long Term: TuringBots Become Autonomous But Stakeholder-Supervised

In the long term (five-plus years), AI will reach high maturity levels in both generation and reasoning. It will address complex problems, break problems into subproblems (i.e., dividuae et impera), make complex inferences, and help structure complex plans by suggesting improvements to human requests. Enterprise governance will also set clear guidelines for TuringBot use (see Figure 7).

TuringBots will be autonomous by performing large batches of work — with human orchestration. Converged-model and data-driven AI with algorithmic AI reasoning will consolidate. TuringBots will scale at the enterprise level, and business will enjoy amazing experiments at a superfast pace. Disrupted, the SDLC will become a practice of the past, as human product managers interact with TuringBots at all levels through language and visual graphic designs (or their preferred description and input language). TuringBots will do all the hard work, and human developers will review, refine, and supervise. The more complex and riskier use cases will be addressed in the longer

term. But the speed at which innovation experiments will run and products are built and deployed for business will be unimaginable.

Figure 7
The Long Term: TuringBots Will Be Developer Leads With Humans As Product Managers



Source: Forrester Research, Inc. Unauthorized reproduction, citation, or distribution prohibited.

Supplemental Material

Companies We Interviewed For This Report

Thank you to the Forrester analysts who contributed to this report: Naveen Chhabra, Christopher Condo, and Janet Worthington.

We also want to thank our clients who contributed to this research: Ruchir Puri (chief scientist, IBM Research), Mark Burnett (head of technology innovation, G-Research), Mathew Lodge (CEO, Diffblue), Ryan Salva (VP of product, GitHub), Igor Kirilenko (chief product officer, Parasoft), Udi Weinberg (director of product management, lifecycle, and portfolio management, OpenText), Yuval Koren (director of R&D, value stream optimization pillar, OpenText), and Gaston Milano (chief technology officer, Globant/GeneXus).

We would also like to thank the individuals from the following companies who generously gave their time during the research for this report.

Accenture

Amazon Web Services

Appvance

Atlassian

CircleCI

Diffblue

Elanco

GeneXus

GitHub

GitLab

Globant

Google Cloud Platform

G-Research

IBM

IBM Consulting

Magic AI

OpenText

Parasoft

Replit

Stack Overflow

Tabnine

Thoughtworks

Tricentis Testim

USPS



We help business and technology leaders use customer obsession to accelerate growth.

FORRESTER.COM

Obsessed With Customer Obsession

At Forrester, customer obsession is at the core of everything we do. We're on your side and by your side to help you become more customer obsessed.

Research

Accelerate your impact on the market with a proven path to growth.

- Customer and market dynamics
- Curated tools and frameworks
- Objective advice
- Hands-on guidance

[Learn more.](#)

Consulting

Implement modern strategies that align and empower teams.

- In-depth strategic projects
- Webinars, speeches, and workshops
- Custom content

[Learn more.](#)

Events

Develop fresh perspectives, draw inspiration from leaders, and network with peers.

- Thought leadership, frameworks, and models
- One-on-ones with peers and analysts
- In-person and virtual experiences

[Learn more.](#)

FOLLOW FORRESTER



Contact Us

Contact Forrester at www.forrester.com/contactus. For information on hard-copy or electronic reprints, please contact your Account Team or reprints@forrester.com. We offer quantity discounts and special pricing for academic and nonprofit institutions.

Forrester Research, Inc., 60 Acorn Park Drive, Cambridge, MA 02140 USA
Tel: +1 617-613-6000 | Fax: +1 617-613-5000 | forrester.com