Paper CD13

Beyond the Code: Evolving from Clinical Programmer to Strategic Partner in Clinical Trials

Anastasiia Khoba, Intego-Group, Kharkiv, Ukraine

ABSTRACT

In today's fast-paced clinical trial landscape, the role of a clinical programmer is evolving far beyond code writing. As data play crucial role for regulatory decisions and trial outcomes, programmers are increasingly expected to contribute not only technical solutions but also strategic insights.

This paper will explore how programmers can grow into influential roles within cross-functional teams by developing leadership, communication, and business-awareness skills as well as how to effectively collaborate with non-programmers' roles to become a strategic partner rather than a task executor.

INTRODUCTION

For many years, clinical programmers have been viewed primarily as "coders"- the behind-the-scenes professionals responsible for building datasets, writing macros, and producing tables, listings, and figures. While this technical work remains essential, the landscape of clinical research itself has changed dramatically. Trials are becoming increasingly complex, data sources more diversified, and regulatory demands more rigorous. At the same time, sponsors are under pressure to deliver the results faster and more cost-effectively.

In this environment, the role of the clinical programmer is expanding. We are no longer "behind the code." Instead, we are becoming **strategic partners** - helping set up study design, anticipating regulatory needs, and driving teams adopt innovative solutions. This transformation is not only reshaping individual careers but also redefining how programming contributes to the success of clinical research.

FROM TECHNICIAN TO STRATEGIC CONTRIBUTOR

Traditionally, programmers were expected to focus on producing deliverables: SDTM and ADaM datasets, statistical outputs, and submission packages. These contributions were viewed as technical support to statisticians and clinicals.

Today, expectations have broadened. Programmers are recognized for their ability to influence trial efficiency and regulatory readiness. Drivers of this change include:

- **Regulatory complexity**: Agencies expect strict compliance with CDISC and evolving standards, where programmers often provide the expertise that ensures submissions are both accurate and efficient [1,2].
- **Adaptive trial designs require quick changes**: Trials often need flexible statistical analysis system and code that can respond to mid-trial changes without compromising data so that it remains accurate, complete, and reliable from the moment it is collected till final reporting.
- **New data sources**: Wearables, real-world data, and patient-reported outcomes bring fresh challenges where programmers' expertise in integration and quality control is invaluable.
- **Need for speed**: In modern clinical research, time has become a critical success factor. The faster sponsors can move from data collection to insight, the faster they can make key decisions - such as dose adjustments, interim analyses, or even regulatory submissions. Programmers play a key role in designing automated, standardized, and real-time workflows that accelerate insights without jeopardizing quality or compliance.

Considering all the above, clinical programmers now influence far more than dataset production. Their expertise has become essential across multiple phases of trial execution - transforming processes, improving efficiency, and ensuring that clinical research teams can deliver results with greater speed, quality, and confidence.

Programmers are often involved in Protocol and CRF review, helping ensure that data collection supports efficient downstream transformation and regulatory compliance from the earliest stages of the study. This proactive involvement reduces rework and enhances overall study quality.

Programming team also work closely with statisticians, co-developing strategies for validation, automation, and reproducibility. By anticipating statistical needs and translating analytical ideas into practical solutions, programmers make the analysis process smoother and more reliable. Furthermore, regulatory submissions, which represent key milestones in each clinical study, are no longer "one-time events." Programmers design metadata-driven pipelines that maintain consistency across studies and prepare sponsors for ongoing, evolving regulatory expectations during the submission process [2]'.

ADVICE 1:

Be open to any challenges. Think of yourself not just as a coder, but as a problem solver, a communicator, and a partner in science.

ADVICE 2:
Learn the full trial process - not just the programming part. Understanding the flow of data from collection to submission helps you build smarter and faster solutions. Stay close to statisticians, data managers, and study leads, and ask questions early.

ADVICE 3:
Be proactive. Don't wait for tasks to be assigned - look for ways to make the process better. Offer suggestions, share lessons learned, and take initiative. The most respected programmers are those who see beyond their own deliverables.

TECHNOLOGY AND INNOVATION LEADERSHIP

The modern clinical programmer's toolbox has expanded far beyond SAS. Innovation is no longer optional - it is expected. Programmers are leading the bringing of new technologies into practice that make trials faster, smarter, and more efficient.
Many programmers now use R and Python, to extend analytical capabilities and enhancing graphics and reporting [3]. Visualization dashboards like R Shiny or Spotfire make it possible to track study progress in near real time, allowing real-time insights into trial status and catching safety signals promptly.
Streamlining processes with standard code libraries and metadata repositories helps build repeatable, reliable processes that reduce manual work and make programmers lives much easier [4]. Artificial intelligence and machine learning are developing extremely fast nowadays and finding their place in programming. They can help identify data issues early, predict enrolment modelling, automate coding tasks, etc.

ADVICE 4:
Keep learning and stay curious. Explore new tools - even if your current project still uses older ones, experimenting with new technology will expand your skills and make you ready for what's next.

ADVICE 5:
Learn how AI and data science are changing the field. The programmers who adapt fastest will be the ones shaping the future of clinical research. Do not view AI as a replacement for your expertise - instead, learn how to make it work for you and enhance your capabilities!

ADVICE 6:
Be open to everything new, join professional communities - they are great places to learn, share, and grow together. At PhUSE, we can regularly see how community contributions accelerate the innovations - whether through shared code, open discussions, or working groups that push the industry forward.

BEYOND TECHNICAL SKILLS: STRATEGIC SOFT SKILLS

Technical ability alone is no longer sufficient for success in today's clinical programming career. To be successful, programmers need to develop soft skills that help them work effectively with others and presenting themselves as thought partners, not just executors. Strong interpersonal skills, influencing stakeholders, business awareness, and problem-solving as equally important as coding ability. Strong communication skills allow to explain technical ideas in a way that everyone - statisticians, clinicians, medical writers, or project managers - can understand. The ability to "translate" complex details builds trust in the team and drives collaboration. Understanding the business context of a study helps programmers see how their work affects timelines, budgets, resource allocation and outcomes which is always highly appreciated by the study team. Problem-solving mindset is another critical aspect of the programmer's routine. Rather than waiting for instructions, strategic programmers think ahead, spot risks early, and suggest practical solutions.

ADVICE 7:
Practice explaining your work simply. Listen carefully to what others need. Learn the language of clinical science and project management. The best programmers are not just great coders - they are trusted partners who can connect technical work to business and scientific goals.

ADVICE 8:
Strengthen your problem-solving mindset by staying curious and analytical. When you face an issue, take time to understand the root cause instead of jumping to quick fixes. Break complex problems into smaller steps, look for patterns in data or processes, and ask "why" more than once. Learn from past challenges -document what worked and what didn't - and share those lessons with your team. Over time, this habit builds confidence, creativity, and the ability to find effective solutions even in unfamiliar situations.

ADVICE 9:
Strive to become the bridge between data, science, and operations. By understanding each area and helping them work together, you position yourself as a trusted advisor and a key contributor to the overall success of the trial. Take time to understand not only the data structure but also the scientific goals of the study and the operational challenges behind it. Proactively share insights from your programming work that may affect decision-making, such as trends in data quality, missing information, or potential analysis risks. Read the protocol carefully, ask questions during team

meetings, and learn how each function - clinical, statistical, data management, and regulatory - uses the outputs you create.

CAREER EVOLUTION AND GROWTH PATHS

The shift from technical coder to strategic partner also opens new career pathways. Many still follow the traditional path from Programmer to Senior Programmer to Lead Programmer, but new roles are emerging - such as Data Strategy Lead, Programming Innovation Lead, or Clinical Data Scientist.

If you are good at industry standards one of the prospective roles might be - Data Strategy Lead – who helps guide organizations in how to leverage data standards and technology to accelerate trials. Another option highly in demand today, for those who are keen on learning and following the trends of industry, is a position of Programming Innovation Lead who is responsible for driving adoption of automation, advanced analytics, and new programming languages. One more possible switch, ideally fitting to specialists with math or medical background, is to become a Clinical Data Scientist – the role that blends statistical programming with data science to extract deeper insights from trial and real-world data.

The opportunities are vast, but they require deliberate investment in upskilling. To prepare for these roles, programmers must expand their knowledge base on an ongoing basis: stay current on regulatory guidance, develop therapeutic expertise to understand the clinical context behind the data, build leadership and project management skills to coordinate across teams effectively. In particular, the PhUSE community offers a unique platform for this growth, enabling knowledge-sharing and collaboration across companies and regions.

ADVICE 10:

Keep building your professional skills continuously. Deepen your understanding of CDISC standards and regulatory guidance so you can design compliant, efficient data strategies across multiple studies. Stay current with updates from FDA, EMA, and CDISC, and learn how standards like SDTM and ADaM support consistency, automation, and traceability in submissions.

ADVICE 11:

Strengthen your knowledge of therapeutic areas, as understanding the science behind the data helps you anticipate analytical needs and support clinical goals. Joining every new study, learn the basics of disease mechanisms, key efficacy and safety endpoints, and how treatments are evaluated there. This knowledge allows you to interpret data in context and contribute more effectively to study design and analysis planning.

ADVICE 12:

Equally important to develop leadership and communication skills. Start small - mentor junior colleagues, take ownership of smaller deliveries, or coordinate cross-functional tasks. Every opportunity to lead, teach, or deliver something independently helps you grow, build confidence, teach responsibility, and sharpen your ability to make decisions. Over time, these smaller steps form the foundation of strong leadership.

FUTURE OUTLOOK

The future of clinical programming is bright and full of opportunities. Automation and AI are far from replacing ambitious programmers, but will continue to take over routine tasks, letting programmers to focus less on repetitive coding and more on strategy, oversight, and innovation.

Trials will become increasingly data-driven, with continuous data flows from many sources. As more diverse data streams are integrated, programmers will become central to designing systems that ensure consistency and quality. The clinical programmer of the future will not be a “back-office coder” but a key partner in clinical development - helping design smarter trials, ensure data quality, and deliver faster insights to support decision-making.

ADVICE 13:

Stay adaptable. Technology will keep changing, but the ability to think critically, communicate clearly, and collaborate effectively will always matter. At the same time, value your industry knowledge and experience - understanding clinical processes and regulatory expectations is just as important as mastering new tools. Embrace change as an opportunity to grow, not a threat to your role.

CONCLUSION

For the last years, the role of the clinical programmer is undergoing a remarkable transformation. Instead of being a technician behind the scenes, programmers now can step beyond the code and redefine themselves as strategic partners who shape study design, anticipate regulatory requirements, and drive innovation. Combine your technical expertise with strategic thinking, and you’ll be ready to shape how data drives success across an organization. The most successful programmers are those who keep learning, share their knowledge, and help others succeed. Remember, no one becomes a strategic leader overnight - it’s the consistent effort, curiosity, and willingness to take initiative that shape you into a trusted professional who inspires others.

At PhUSE, every time we see this transformation in action -in the knowledge shared, the tools developed, and the passion of our community to push the boundaries of what programming can achieve.

The journey forward is clear. Clinical programmers are not just coding -we are collaborating, innovating, and enabling

scientific progress. Together, as a global community, we can continue to evolve **beyond the code** and help deliver therapies to patients faster and with greater impact.

REFERENCES

1. CDISC. Clinical Data Interchange Standards Consortium Standards & Implementation Guides. Available at: https://www.cdisc.org/standards
2. U.S. Food and Drug Administration (FDA). Study Data Technical Conformance Guide. Latest version. Available at: https://www.fda.gov
3. R Consortium. R in Regulatory Submissions (R Submissions Working Group). Available at: https://rconsortium.github.io/submissions-wg/
4. PhUSE Working Group. Available at: https://phuse.global/Working_Groups

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Author Name: Anastasiia Khoba
Company: Intego-Group
Email: Anastasiia.khoba@intego-group.com