

Thursday 6th July 2023, The Foundling Museum The Spreadsheet Crisis: Regaining Control

08:30 – 09:30 Arrival, Registration & Coffee

09:30 – 10.15 [Session 1 \(Keynote\)](#)

Bringing generative AI to the Excel grid: from research to practice (Keynote), Andrew D. Gordon and Jack Williams, Calc Intelligence Microsoft Research

10.15 – 11.00 [Session 2](#)

Reducing Errors in Excel Models with Component-Based Software Engineering, Craig Hatmaker

11.00 – 11.30 **Coffee Break**

11.30 – 12.15 [Session 3](#)

A Use Case-Engineering Resources Taxonomy for Analytical Spreadsheet Models, Tom Grossman and Vijay Mehrotra, University of San Francisco

12.15 – 13:00 [Session 4](#)

How Beaufort, Neumann, and Gates met? Subject integration with spreadsheeting, Maria Csernoch and Julia Csernoch, University of Debrecen, Faculty of Informatics

13.00 – 14:00 **Lunch**

14:00 – 14:45 [Session 5](#)

Excel as a Turing-complete Functional Programming Environment, Peter Bartholomew, MDAO Technologies Ltd

14:45 – 15:30 [Session 6](#)

ChatGPT and Excel – trust, but verify, Patrick O'Beirne, Systems Modelling Ltd

15.30 – 16:15 [Session 7](#)

Experimenting with ChatGPT for Spreadsheet Formulae Generation: The Risks of AI Generated Spreadsheets, Simon Thorne, Cardiff Metropolitan University

16:15 – 17:00 **Q&A and Tea Break**

17:00 -- **Conference Close**

AGM to discuss future direction of EuSpRIG.

Bringing generative AI to the Excel grid: from research to practice (Keynote)

Andrew D. Gordon and Jack Williams,
Calc Intelligence Microsoft Research
Team link: <https://aka.ms/CalcIntel>

ABSTRACT

Learn how to transform your textual data within the Excel grid using the new `LABS.GENERATIVEAI()` function. It empowers Excel users with OpenAI's pre-trained language models. It's part of the new add-in *Excel Labs, a Microsoft Garage project* that also delivers new formula editing and reuse features, including easy editing of LAMBDA's. We'll showcase the productivity benefits of using `LABS.GENERATIVEAI()` in conjunction with LAMBDA-defined functions. These features build on our research in the Calc Intelligence team at Microsoft Research in Cambridge and arise from a long-term partnership with Excel

Reducing Errors in Excel Models with Component-Based Software Engineering

Craig Hatmaker

ABSTRACT

Model errors are pervasive and can be catastrophic. We can reduce model errors and time to market by applying Component-Based Software Engineering (CBSE) concepts to Excel models. CBSE assembles solutions from pre-built, pre-tested components rather than written from formulas. This is made possible by the introduction of LAMBDA. LAMBDA is an Excel function that creates functions from Excel's formulas. CBSE-compliant LAMBDA functions can be reused in any project just like any Excel function. They also look exactly like Excel's native functions such as `SUM()`. This makes it possible for even junior modelers to leverage CBSE-compliant LAMBDA's to develop models quicker with fewer errors.

A Use Case-Engineering Resources Taxonomy for Analytical Spreadsheet Models

Tom Grossman, Vijay Mehrotra
University of San Francisco

ABSTRACT

This paper presents a taxonomy for analytical spreadsheet models. It considers both the use case that a spreadsheet is meant to serve, and the engineering resources devoted to its development. We extend a previous three-type taxonomy, to identify nine types of spreadsheet models, that encompass the many analytical spreadsheet models seen in the literature. We connect disparate research literature to distinguish between an “analytical solution” and an “industrial-quality analytical spreadsheet model”. We explore the nature of each of the nine types, propose definitions for some, relate them to the literature, and hypothesize on how they might arise. The taxonomy provides guidance for where various spreadsheet development guidelines are most useful, provides a lens for viewing spreadsheet errors and risk, and offers a structure for understanding how spreadsheets change over time. This taxonomy opens the door to many interesting research questions, including refinements to itself.

How Beaufort, Neumann, and Gates met? Subject integration with spreadsheeting

Maria Csernoch, Julia Csernoch
University of Debrecen, Faculty of Informatics

ABSTRACT

Computational thinking should be the fourth fundamental skill, along with reading, writing, and arithmetic (3R). To reach the level where computational thinking skills, especially digital problem solving have their own schemata, there is a long way to go. In the present paper, a novel approach is detailed to support subject integration and building digital schemata, on the well-known Beaufort scale. The conversion of a traditional, paper-based problem and a data retrieval process are presented within the frame of a Grade 8 action research study. It is found that both students’ content knowledge and their digital skills developed more efficiently than in traditional course book and decontextualized digital environments. Furthermore, the method presented here can be adapted to any paper-based problems whose solutions would be more effective in a digital environment and which offer various forms for building schemata both in the subject matter and informatics.

Excel as a Turing-complete Functional Programming Environment

Peter Bartholomew
MDAO Technologies Ltd

ABSTRACT

Since the calculation engine of Excel was the subject of a major upgrade to accommodate Dynamic Arrays in 2018 there have been a series of seismic changes to the art of building spreadsheet solutions. This paper will show the ad-hoc end user practices of traditional spreadsheets can be replaced by radically different approaches that have far more in common with formal programming. It is too early to guess the extent to which the new functionality will be adopted by the business and engineering communities and the impact that may have upon risk. Nevertheless, some trends are emerging from pioneering work within the Excel community which we will discuss here.

ChatGPT and Excel – Trust, but Verify

Patrick O'Beirne
Systems Modelling Ltd

ABSTRACT

This paper adopts a critical approach to GPT-4, showing how its huge reach makes it a useful tool for people with simple requirements but a bad, even misleading guide to those with more complex problems which are more rarely present in the training data and even more rarely have straightforward solutions.

It concludes with a practical guide for how to add an Excelscript button, with system and user prompts, to the GPT-4 API into the Excel desktop environment, supported by a blog post giving the technical details for those interested.

Experimenting with ChatGPT for Spreadsheet Formulae Generation: The Risks of AI Generated Spreadsheets

Simon Thorne
Cardiff Metropolitan University

Abstract

Large Language Models (LLM) have become sophisticated enough that complex computer programs can be created through interpretation of plain English sentences and implemented in a variety of modern languages such as Python, Java Script, C++ and Spreadsheets. These tools are powerful and relatively accurate and therefore provide broad access to computer programming regardless of the background or knowledge of the individual using them. This paper presents a series of experiments with ChatGPT to explore it's ability to produce valid spreadsheet formulae and related computational outputs in situations where ChatGPT has to deduce, infer and problem solve the answer. The results show that in certain circumstances, ChatGPT can produce correct spreadsheet formulae with correct reasoning, deduction and inference. However, when information is limited, uncertain or the problem is too complex, the accuracy of ChatGPT breaks down as does its ability to reason, infer and deduce. This can also result in false statements and "hallucinations" that all subvert the process of creating spreadsheet formulae.