RESEARCH ARTICLE SUMMARY

CHEMISTRY

Organic synthesis in a modular robotic system driven by a chemical programming language

Sebastian Steiner, Jakob Wolf, Stefan Glatzel, Anna Andreou, Jarosław M. Granda, Graham Keenan, Trevor Hinkley, Gerardo Aragon-Camarasa, Philip J. Kitson, Davide Angelone, Leroy Cronin*

INTRODUCTION: Outside of a few well-defined areas such as polypeptide and oligonucleotide chemistry, the automation of chemical synthesis has been limited to large-scale bespoke industrial processes, with laboratory-scale and discovery-scale synthesis remaining predominantly a manual process. These areas are generally defined by the ability to synthesize complex molecules by the successive iteration of similar sets of reactions, allowing the synthesis of products by the automation of a relatively small palette of standardized reactions. Recent advances in areas such as flow chemistry, oligosaccharide synthesis, and iterative cross-coupling are expanding the number of compounds synthesized by automated methods. However, there is no universal and interoperable standard that allows the automation of chemical synthesis more generally.

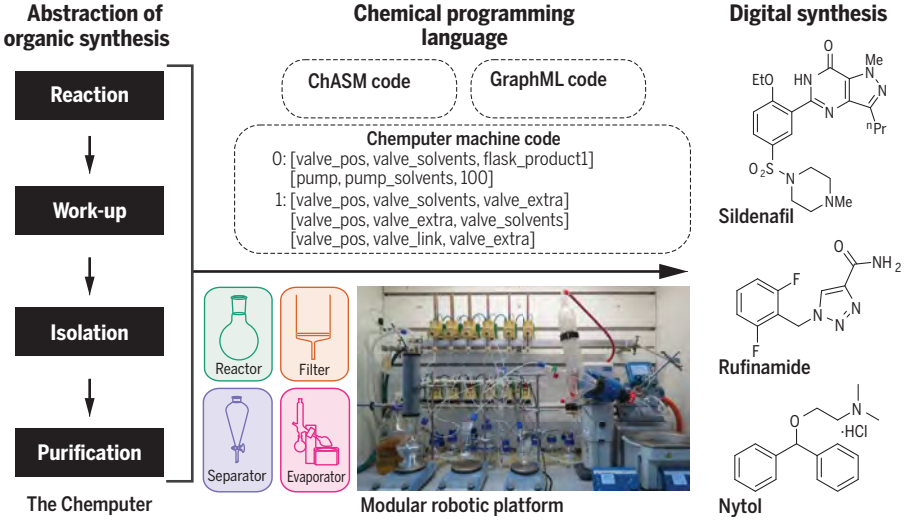
RATIONALE: We developed a standard approach that mirrors how the bench chemist works and how the bulk of the open literature is reported, with the round-bottomed flask as the primary reactor. We assembled a relatively small array of equipment to accomplish a wide variety of different syntheses, and our abstraction of chemical synthesis encompasses the four key stages of synthetic protocols: reaction, workup, isolation, and purification. Further, taking note of the incomplete way chemical procedures are reported, we hypothesized that a standardized format for reporting a chemical synthesis procedure, coupled with an abstraction and formalism linking the synthesis to physical operations of an automated robotic platform, would yield a universal approach to a chemical programming language. We call this architecture and abstraction the Chemputer.

RESULTS: For the Chemputer system to accomplish the automated synthesis of target molecules, we developed a program, the Chempiler, to produce specific, low-level instructions for modular hardware of our laboratory-scale synthesis robot. The Chempiler takes information about the physical connectivity and composition of the automated platform, in the form of a graph using the open-source GraphML format, and combines it with a hardware-independent scripting language [chemical assembly (ChASM) language], which provides instructions for the machine operations of the automated platform. The Chempiler software allows the ChASM code for a protocol to be run without editing on any unique hardware platform that has the correct modules for the synthesis. Formalization of a written synthetic scheme by using a chemical descriptive language (XDL) eliminates the ambiguous interpretation of the synthesis procedures. This XDL scheme is then translated into the ChASM file for a particular protocol. An automated robotic platform was developed, consisting of a fluidic backbone connected to a series of modules capable of performing the operations necessary to complete a synthetic sequence. The backbone allows the efficient transfer of the required chemicals into and out of any module of the platform, as well as the flushing and washing of the entire system during multistep procedures in which the modules are reused multiple times. The modules developed for the system consist of a reaction flask, a jacketed filtration setup capable of being heated or cooled, an automated liquid-liquid separation module, and a solvent evaporation module. With these four modules, it was possible to automate the synthesis of the pharmaceutical compounds diphenhydramine hydrochloride, rufinamide, and sildenafil without human interaction, in yields comparable to those achieved in traditional manual syntheses.

ON OUR WEBSITE

Read the full article at http://dx.doi.org/10.1126/science.aav2211

CONCLUSION: The Chemputer allows for an abstraction of chemical synthesis, when coupled with a high-level chemical programming language, to be compiled by our Chempiler into a low-level code that can run on a modular standard robotic platform for organic synthesis. The software and modular hardware standards permit synthetic protocols to be captured as digital code. This code can be published, versioned, and transferred flexibly between physical platforms with no modification. We validated this concept by the automated synthesis of three pharmaceutical compounds. This represents a step toward the automation of bench-scale chemistry more generally and establishes a standard aiming at increasing reproducibility, safety, and collaboration. ■



Abstraction of organic synthesis to the Chemputer software and hardware standards. The Chemputer system can perform multistep organic synthesis protocols on the bench by using a modular system with hardware and software standards.

The list of author affiliations is available in the full article online.
*Corresponding author. Email: lee.cronin@glasgow.ac.uk
Cite this article as S. Steiner *et al.*, *Science* **363**, eaav2211 (2019). DOI: 10.1126/science.aav2211

RESEARCH ARTICLE

CHEMISTRY

Organic synthesis in a modular robotic system driven by a chemical programming language

Sebastian Steiner[1*], Jakob Wolf[1], Stefan Glatzel[1], Anna Andreou[1], Jarosław M. Granda[1], Graham Keenan[1], Trevor Hinkley[1], Gerardo Aragon-Camarasa[1,2], Philip J. Kitson[1], Davide Angelone[1], Leroy Cronin[1]†

The synthesis of complex organic compounds is largely a manual process that is often incompletely documented. To address these shortcomings, we developed an abstraction that maps commonly reported methodological instructions into discrete steps amenable to automation. These unit operations were implemented in a modular robotic platform by using a chemical programming language that formalizes and controls the assembly of the molecules. We validated the concept by directing the automated system to synthesize three pharmaceutical compounds, diphenhydramine hydrochloride, rufinamide, and sildenafil, without any human intervention. Yields and purities of products and intermediates were comparable to or better than those achieved manually. The syntheses are captured as digital code that can be published, versioned, and transferred flexibly between platforms with no modification, thereby greatly enhancing reproducibility and reliable access to complex molecules.

The automation of chemical synthesis is currently expanding, and this is driven by the availability of digital labware. The field currently encompasses areas as diverse as the design of new reactions (*1*), chemistry in reactionware (*2*), reaction monitoring and optimization (*3*, *4*), flow chemistry (*5*) for reaction optimization and scale up, and even full automation of the synthesis laboratory (*6*). Established technologies such as the automated synthesis of peptides (*7*) and oligonucleotides (*8*), flow chemistry (*9*), and high-throughput experimentation (*6*) are mainstays of modern chemistry, whereas emerging technologies such as automated oligosaccharide synthesis (*10*) and iterative cross-coupling (*1*) have the potential to further transform the chemical sciences. Each of these examples, however, relies on a distinct protocol, as no current digital automation standard exists for computer control of chemical reactions (*11*). Hence, no general programming language is available for chemical operations that can direct the synthesis of organic compounds on an affordable, flexible, modular platform accessible to synthetic chemists and that could, in principle, encompass all synthetic organic chemistry. This situation is comparable to the era before digital programmable computers, when existing computational devices were fixed to a dedicated problem.

A generalized approach to automating chemical synthesis would be beneficial because making compounds is one of the most labor-intensive branches of chemistry, requiring manual execution of a range of unit operations such as reagent mixing, liquid-liquid extractions, or filtrations. Despite the modular nature of the operations, standardization and automation are still severely limited. Furthermore, the ambiguous way in which synthetic protocols are communicated has contributed to a mounting reproducibility crisis (*12*). Syntheses are reported in prose, omitting many details explaining exactly how operations were carried out and making many assumptions about the skill level of the researcher repeating the process. We hypothesized that a more standardized format for reporting a chemical synthesis procedure, coupled with an abstraction and formalism linking the synthesis to the physical operations, could yield a universal approach to a chemical programming language; we call this architecture and abstraction the Chemputer.

In developing the Chemputer platform, we wanted to build on the 200 or more years of chemical literature and the experience of the many thousands of bench chemists active in the world today in a way that would naturally lead to a standardization embodied in a codified standard recipe, or chemical code, for molecular synthesis. For this to be possible, it was essential that the approach mirror the way the bench chemist works. As our key reaction module for implementation of the Chemputer, we selected the round-bottomed flask for batch synthesis, with well-defined inputs and outputs, because most protocols already published for complex molecule synthesis rely on this type of apparatus. Next, we identified the four key stages of a synthetic protocol from our abstraction: reaction, workup, isolation, and purification. These stages can be subdivided into several unit operations, which in turn are implemented in a specific automated platform.

Developing code for chemistry

By developing control software as well as hardware modules for laboratory-scale synthesis that can be automatically cleaned and reused in subsequent reaction steps, we were able to define a process for combining individual unit operations into full, multistep syntheses to produce desired products autonomously (Fig. 1A). For the Chemputer to operate as shown in Fig. 1B, the states of the inputs, reactor, and outputs must be defined and controlled programmatically. We therefore created a Chempiler, which is a program to produce specific low-level instructions for the modules that constitute the Chemputer architecture. The Chempiler can run commands used to control the modular platform from our abstraction layer, so that a typical written scheme can be turned into a specific code to run the modules with ease. Every module was then designed with drivers for the device or equipment and a standardized application programming interface (API) exposing the instruction set to the Chempiler (Fig. 2). The use of a program coupled to the Chemputer architecture allows users to directly run published syntheses without reconfiguration, provided the necessary modules and drivers are present within their system. Thus, the practical implementation of the Chemputer architecture converts digital code to chemical synthesis operations in accordance with the standard protocol of a chemical reaction incorporating the four processes listed in Fig. 1C. To prevent the need for manual reconfiguration, the physical modules and their connections and representation are stored in memory as a directed graph, which allows knowledge and control of their states in real time.

The physical routing that links the connected modules is described in a graph by using an open-source format, GraphML, which allows the Chempiler to find paths between a source flask and a target flask, as well as address devices such as hot plate stirrers on the basis of the vessel they are connected to. GraphML is an open-standard, extensible markup language (XML)–based exchange format for graphs (*13*) [example graphs used in the syntheses described herein can be found in data S1 (GraphML files) in the supplementary materials (SM)]. Synthetic procedures are codified by using a scripting language called Chemical Assembly (ChASM), which provides instructions for all currently implemented machine operations and supports the definition of functions and variables. To develop the ChASM code, we built a standard procedure, starting with a written synthetic scheme that is formalized by using a chemical descriptive language (XDL).

[1]School of Chemistry, The University of Glasgow, Glasgow G12 8QQ, UK. [2]School of Computer Science, The University of Glasgow, Glasgow G12 8QQ, UK.
*Present address: Department of Chemistry, University of British Columbia, Vancouver, British Columbia V6T 1Z1, Canada.
†Corresponding author. Email: lee.cronin@glasgow.ac.uk

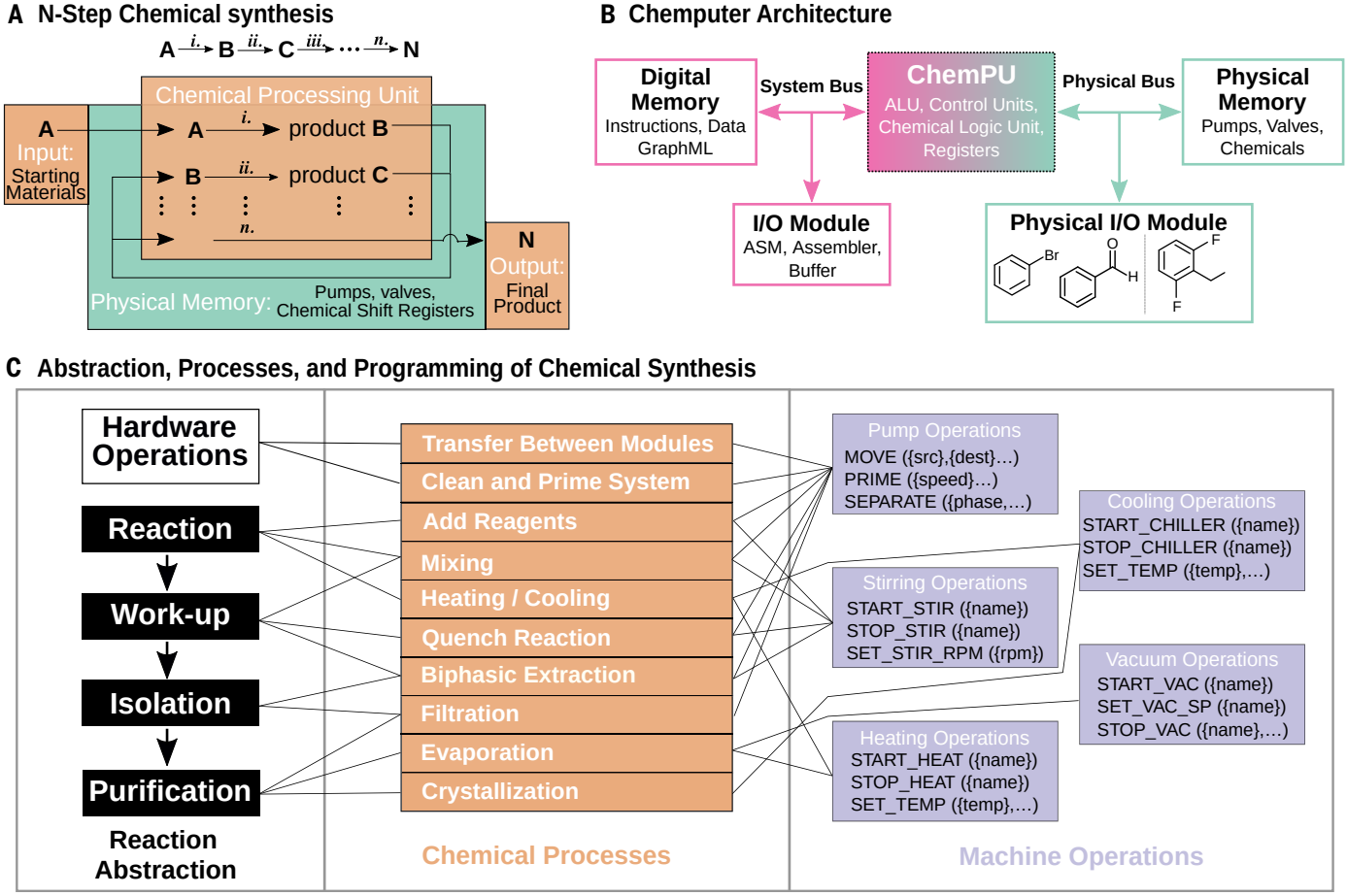


Fig. 1. Operating principles of the Chemputer. (**A**) Schematic representation of a stepwise chemical synthesis, formalizing the reagents as inputs and products as output. (**B**) Diagram outlining the computing architecture of the Chemputer. ALU, arithmetic logic unit; I/O, input/output; ASM, assembly language. (**C**) The abstraction of chemical synthesis allowing the development of an ontology that can be universally programmed by using a machine. Similar to digital computers, the Chemputer has a memory and a bus system, but these are both digital and physical. By considering chemical reagents and products in a memory bus, it is possible to break the process of complex molecule synthesis into steps or cycles that can be run using the physical hardware.

The XDL has the advantage of eliminating ambiguous interpretation of the synthesis procedures by explicitly and systematically listing all required information without making any assumptions or inferences (example ChASM files used in the syntheses described herein can be found in data S2).

To control the chemistry, the Chempiler was designed to accept ChASM commands such as "start stirring the reactor," find the module in question in the GraphML definition, and schedule the execution by using the appropriate low-level instructions. The modular Chemputer components thereby constitute a versatile and interoperable architecture for chemical synthesis. Also, a given ChASM file could be run on many different platform instances with hardware of different makes and models connected in different ways. If the required unit operation modules are present and all required reagents and solvents are provided, the hardware-agnostic ChASM code can be freely paired with a system-specific GraphML file to synthesize the same product on another, different platform without reoptimization.

During the development of the process and programming language, we found it helpful to visualize the workflow one would follow when manually reproducing the procedure. Writing the XDL and converting it to ChASM then become intuitive even for users with no programming experience. Many operations are repetitive enough to be defined as functions that can be reused many times. For instance, we have defined several functions such as "evaporate to dryness" or "add reagents and heat up to x°C," and we expect that this library can be greatly expanded with future use. Once the ChASM file is completed, a graph of the platform is drafted, following a set of rules detailed in the SM. To validate the ChASM and the graph, we implemented two simulation modes in the Chempiler. First, all operations are performed as they would be on the real system, but instead of commands being issued to physical devices, the commands are logged to a text file. Next, the system simulates the process without reagents as a dry run. The simulations flag any potential issues, such as syntax errors in the XDL representation and the ChASM, inconsistencies in the graph representation, impossible operations, or overfilling of vessels. Once the simulations run without errors, the user can load the reagents and solvents in accordance with the graph and start the synthesis. The Chempiler also has a break-point command option for additional safety and compatibility checks as needed.

Modular synthesis platform

To produce a physical platform that could implement our architecture and achieve the syntheses outlined below, we had to step away from thinking about reactions and rather adopt a process-centered way of thinking. Although the 20 most commonly used reaction classes in drug discovery (*14*) span a wide range of chemistries, from amide bond formation to Buchwald-Hartwig couplings, experimentally most of them simply involve the mixing of several reagents in the

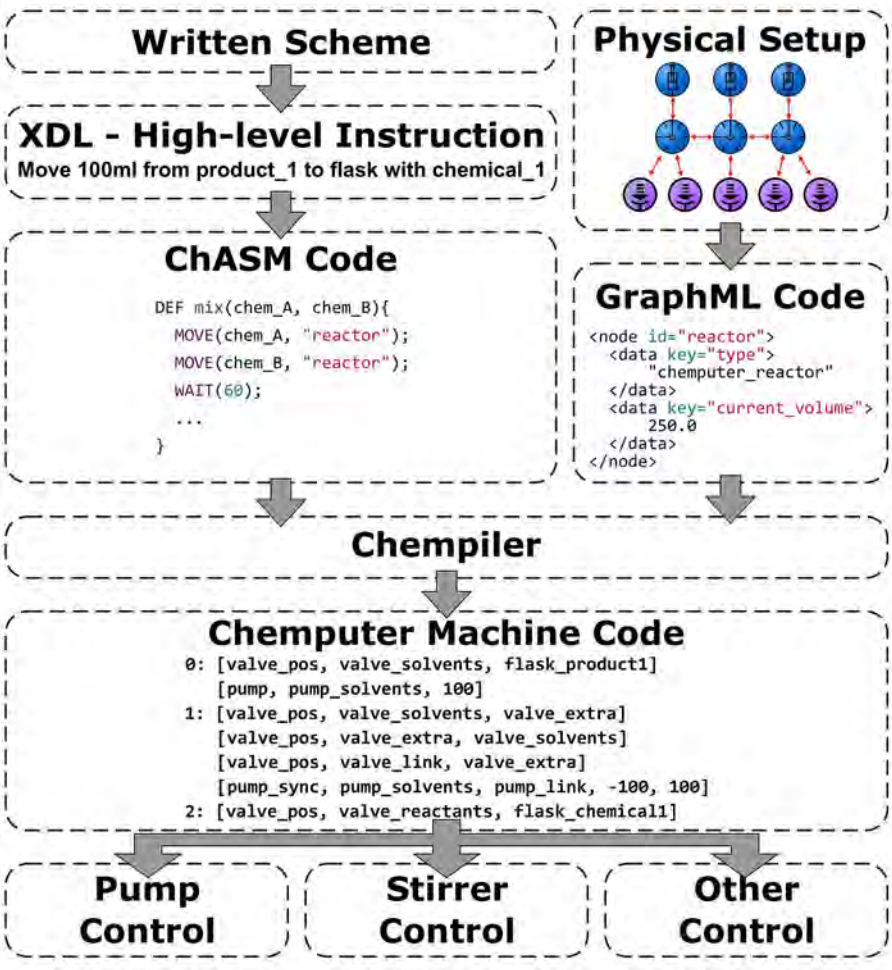


Fig. 2. Diagram showing the flow of information in the synthesis platform. To translate experimental steps into a set of pump movements or hot plate stirrer commands, the user provides the synthesis instructions in a text format similar to a written protocol. The graphical representation of the synthesis platform, which contains all necessary information about the fluidic connectivity, is represented by using GraphML. The written scheme from a published procedure or lab book entry is translated into a series of ChASM commands. Both the ChASM and the GraphML files are passed to the Chempiler. The Chempiler command dispatcher then uses those two pieces of information to control the physical labware through the respective device drivers, effectively executing the synthesis.

correct order, often under heating or cooling. These processes are typically followed by a workup and a purification technique such as distillation, recrystallization, or chromatography. We therefore concluded that a synthesis platform capable of performing the unit operations of mixing under heating or cooling, liquid-liquid separation, filtration, and solvent evaporation could in principle perform a large fraction of all organic syntheses and embody our abstraction of chemical synthesis (Fig. 1C). As those unit operations do not always occur in the same order, especially where multistep syntheses are concerned, a flexible means of moving material between the modules was required. To that end, we built the physical architecture around a fluidic backbone consisting of a series of syringe pumps and six-way selection valves (Fig. 3). The appeal of this design is that it is expandable: The user can always add more pump-valve units (backbone units) to the ends of the backbone. Material can be moved between modules in an arbitrary order, including multiple uses of the same module at different points in the sequence. The process to transfer liquid from a port on the backbone to another backbone is described in the SM (fig. S8). The pump on the source unit aspirates the appropriate amount, then the valve on the source unit and the adjacent unit switch to the bridge, and the source pump and the adjacent pump move simultaneously to transfer the liquid contents from the source syringe to the next syringe. This process is repeated, and the liquid is moved along the backbone until it reaches the destination unit, which in turn dispenses it to the destination port. Surplus ports on the six-way valves accommodate solvents and reagents, and additional backbone units may be dedicated entirely to supplying chemicals. This is in contrast with even the most advanced flow chemical setups to date (*15*), which have to be physically reconfigured for every new synthesis. In flow, the number and sequence of unit operation modules must match the number and sequence of required unit operations, whereas in our system the ability to address modules independently and reuse them as required removes the need for physical reconfiguration. This is achieved by using GraphML, which gives a complete description of the connectivity, allowing the pumps and valves to be dynamically reconfigurable resources, and facilitates the movement of solvents and solutions from reagent flasks to the various components required for a given synthesis step. To ensure modularity, we designed our own pumps and valves, controlled and powered by a single Ethernet cable plugged into each item, powered from a network switch placed next to the fume hood (see fig. S11).

In this work, we developed modules for the unit operations of mixing, filtration, liquid-liquid separation, evaporation, and chromatographic separation, which are all key to the practical implementation of our abstraction of a chemical reaction. Detailed descriptions of the individual modules can be found in figs. S34 to S44 and accompanying supplementary text. However, both the physical architecture and the architecture of the Chempiler (vide supra) were specifically designed to allow for future addition of other modules, enabling automation of even more reactions.

The reactor (fig. S34) consisted of a commercially available, two-necked, pear-shaped flask equipped with an air condenser. We decided to use common laboratory glassware rather than a jacketed reactor vessel both for simplicity and to lower the barrier for reproducing the setup. A pear-shaped flask was chosen over the more common round-bottomed flask to accommodate a wider range of reaction volumes. Heating and stirring were accomplished by using a computer-controllable stirrer hot plate and a custom-manufactured aluminum heating block for the pear-shaped flask. A 0.32-cm-outer-diameter polytetrafluoroethylene (PTFE) tube was held in place by a ground glass joint–to–GL18 thread adapter with a GL18 screw cap and insert so that its end reached the bottom of the flask. A slight argon overpressure was maintained by the inert gas system (see SM). For the sildenafil synthesis, cooling of the reactor was required, and we used a recirculation chiller with a temperature range of −30 to 160°C, giving precise temperature control of the reactor.

The liquid-liquid separation, one of the most common isolation techniques, was also the most challenging task to automate in a robust fashion. Whereas in continuous flow there are solutions using membrane technology (*16*), we found that commercially available hydrophobic frits are usually designed to be single use and therefore lack long-term reliability. We attempted to use

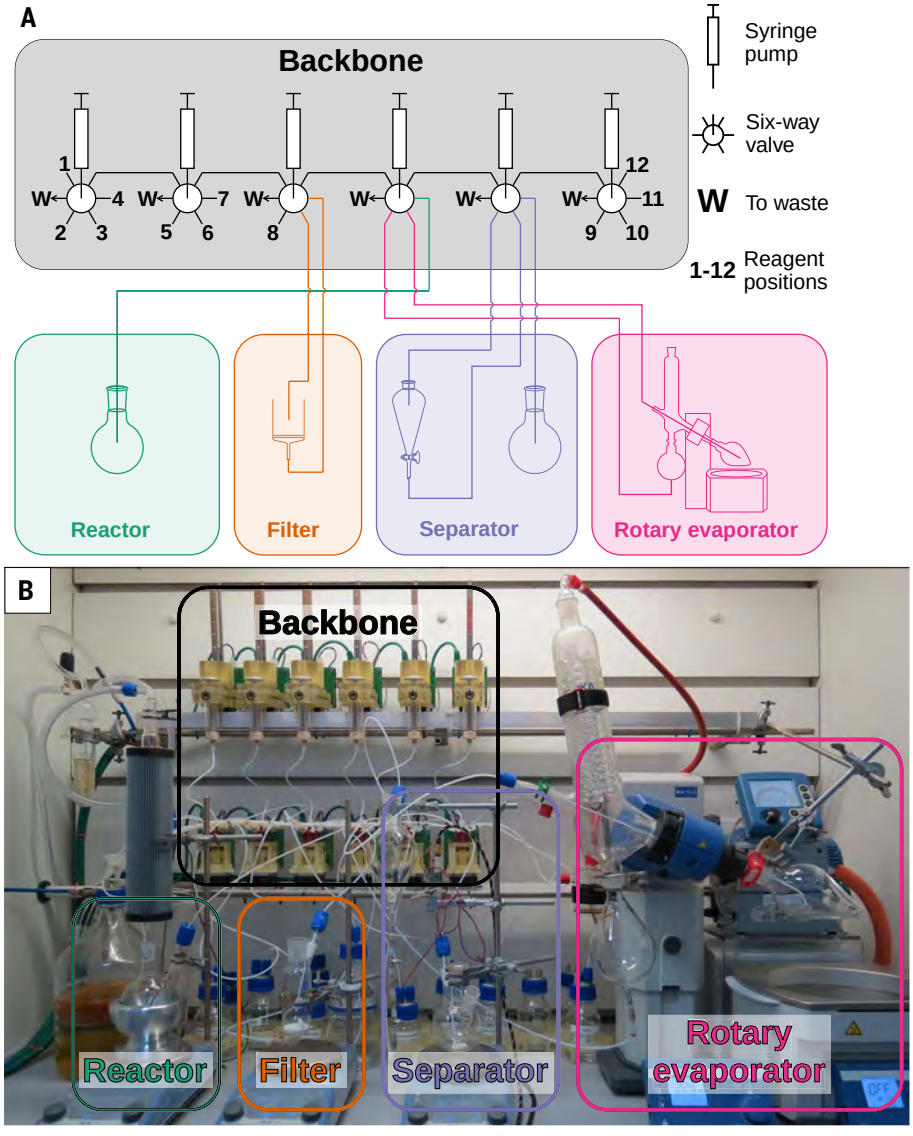


Fig. 3. Physical implementation of the synthesis platform. (**A**) Schematic representation of the Chemputer, highlighting modules used for four commonly encountered unit operations. The lines represent fluidic connections. (**B**) Photograph of one Chemputer setup used in this work. The various modules are highlighted in correspondence to the schematic.

a modified separating funnel and computer vision to directly replicate the way a human chemist would perform liquid-liquid extractions. However, we found that solutions using either a colored floater (*17*) or direct recognition of the phase boundary (*18*) worked well in a test environment but were otherwise unreliable. In real syntheses, imperfections such as poor phase separation, strongly colored or cloudy solutions, or unusual extraction solvents often lead to complete failure of the image recognition algorithms. We therefore abandoned the computer vision for a sensor-based approach. Initially, we investigated optical and capacitive sensors because they do not require direct contact with the medium. Unfortunately, those sensors also performed poorly in some cases, so ultimately, we built a conductivity sensor from two pieces of stainless-steel tubing inserted into the flow path (see SM). This sensor reliably detected phase boundaries in all test cases and enabled us to perform separations in a robust fashion. The sensor was connected to a custom-made separating funnel with a B45 ground glass joint at the top and two B14 side arms (fig. S36). Instead of a stopcock, it had a glass ¼-28 UNF (Unified fine) male threaded connector (where ¼ indicates a major diameter of 0.25 inches, or 6.35 mm, and 28 indicates a pitch of 28 threads per inch) fitted to the bottom. An Arduino Due was used to read the sensor via a simple voltage divider circuit. The top inlet tube was suspended by a ground glass joint–to–GL18 thread adapter with a GL18 screw cap and insert. To facilitate efficient extractions through thorough mixing, a computer-controlled overhead stirrer was fitted above the separator.

To perform liquid-liquid extractions, the mixture was pumped into the separator through the top inlet, or in the case of a wash, the washing solvent was added through either the top or bottom port, depending on whether it constituted the top or bottom layer of the biphasic mixture. The two layers were then stirred vigorously with the overhead stirrer, and this step was followed by a period of settling under very slow (50 rpm) stirring. Next, the bulk of the lower phase was usually transferred to the target vessel to speed up the process. The volume moved was determined empirically for every separation. The actual separation commenced with withdrawing the dead volume of the sensor and tubing from the bottom port. The volume removed was dispensed into the lower-phase target vessel. Then a sensor reading was taken and compared against a reference value. If the reading was lower than the reference, the lower phase was assumed to be organic; otherwise, the lower phase was assumed to be aqueous. One milliliter was then transferred to the lower-phase target. Another sensor reading was taken and compared against a predefined threshold value. This threshold depended on whether the lower phase was aqueous or organic. Either way, if the sensor reading was outside the threshold, it was concluded that a phase change had been detected. If not, another milliliter was withdrawn and the cycle continued until a phase change was detected. Then the dead volume of the sensor and tubing was transferred to the lower-phase target vessel. If the upper-phase target was specified as the separator, the process was concluded. Otherwise, the upper phase was withdrawn as well and transferred to the target vessel.

For the solvent evaporation, a computer-controllable rotary evaporator was modified by routing a piece of PTFE tubing through the vapor duct into the evaporation flask to pump product into and out of the flask (fig. S37). The receiver flask was fitted with a glass ¼-28 UNF male threaded connector and a PTFE tube, allowing it to be emptied in situ. One complication was that after distillation at reduced pressure, upon venting, oily products were forced back into the tube reaching into the evaporation flask. This problem was solved by affixing a magnetic stirrer bar to the tube by using PTFE shrink wrap (fig. S38). A strong magnet was then positioned in such a way that it would attract the tube and lift it out of the product, allowing the system to be vented. When the flask was lowered into the heating bath, the tube would be released and drop, thereby allowing product to be withdrawn. Solvent evaporation started with pumping the solution to be evaporated into the distillation flask of the rotary evaporator. A cartridge filled with molecular sieves could be switched into the flow path by two six-way selection valves, allowing for the solution to be dried before evaporation. The flask was then lowered into the heating bath, and the rotation was turned on. The vacuum pump was started, lowering the pressure to 900 mbar to degas the solution and avoid excessive foaming later on. The heating bath and the recirculation chiller servicing the condenser were switched on, the target temperatures were

set, and execution of the script was suspended until the target temperatures were reached. The vacuum set point was then changed to the target distillation pressure, and the vacuum pump speed was adjusted according to the solvent, to avoid bumping. Then the execution of script was suspended for a user-defined amount of time to allow the main distillation to finish. After the distillation was complete, the flask was lifted, which caused the inlet tube to be attracted by the magnet (fig. S38), lifting it out of the remaining solution. The vacuum pump was subsequently stopped, and the vacuum was vented. A user-defined amount of distillate was removed from the distillate flask and discarded. The parameters (pressure, timings, and volumes) were always chosen empirically, either through experimental trial and error or from prior experiences with the system.

At this point, we found that proceeding directly to drying the product under maximal vacuum would often lead to a few milliliters of residual solvent distilling over, which decreased the drying efficiency. Thus, the flask was lowered back into the bath and the vacuum pump was set to maximum power for 2 min, drawing out any residual solvent. The sequence of raising the flask, venting the vacuum, and emptying the distillate flask was then repeated. Next, the flask was lowered once again, the vacuum pump was set to maximum power and started, and the cooling of the condenser was switched off. The product was then dried for a user-defined amount of time. After the drying was complete, the flask was lifted once more, the vacuum was vented, and the rotation and heating bath were switched off.

The filtration unit consisted of a custom-made, jacketed, sintered glass Büchner funnel (made in-house; see SM) fitted with a B29 ground glass joint at the top, two B14 side arms, and a glass ¼-28 UNF male threaded connector at the bottom. The top inlet tube was suspended by a ground glass joint–to–GL18 thread adapter with a GL18 screw cap and insert, whereas the bottom outlet tube was connected to the threaded connector with a straight union piece. Stirring was accomplished by a computer-controllable overhead stirrer.

To allow efficient drying of the precipitate, the bottom outlet of the filter was connected to the central inlet of a six-way valve. One outlet of that valve was then connected to the backbone, and another outlet was connected to the laboratory vacuum system via a Woulff bottle. This allowed the user to switch the filter bottom between the backbone (for liquid addition or withdrawal) and vacuum (for drying). The whole platform could be cleaned automatically by pumping suitable washing solvents into the modules. This cleaning cycle would return the system to its initial state, ready for the next reaction stage. Consequently, preparing the system for another synthesis was simply a matter of swapping the reagent bottles.

Proof-of-principle automated syntheses of three drugs

To highlight the power of this approach, we chose three targets: diphenhydramine hydrochloride (**6**) (Fig. 4), rufinamide (**10**) (Fig. 5), and sildenafil (**17**) (Fig. 6). The process of digitizing a synthesis always starts with a traditional, written scheme such as an experimental procedure from a publication or a lab book entry. We chose three published syntheses, all replicated manually to establish benchmark yields and purities for comparison with the automated runs.

Diphenhydramine hydrochloride is an ethanolamine derivative used as an antihistamine and a mild sleep aid. It is marketed as Nytol in the United Kingdom and as Benadryl in the United States. The synthesis is a four-step sequence starting with a Grignard reaction. Rufinamide is a triazole derivative used as an anticonvulsant to treat various seizure disorders, and its synthesis is a relatively simple process to automate. Sildenafil is prescribed to treat erectile dysfunction and is best known under the brand name VIAGRA; its industrial synthesis route (*19*) features a chlorosulfonation with highly aggressive chlorosulfonic acid and thionyl chloride. We reasoned that successful automated handling of those aggressive reagents would demonstrate the versatility and robustness of the system, as well as highlighting the safety benefit arising from automating dangerous procedures.

After reproducing the synthesis of diphenhydramine hydrochloride (**6**) (*20*) manually, we made some small modifications and started the process on the platform. The synthesis commenced with Grignard reagent formation, for which the reactor was manually charged with dried magnesium grit. All the other required reagents and solvents were loaded into 100- or 250-ml standard GL45 bottles, and all nonaqueous storage bottles were purged with argon and stored under positive pressure. All the operations described were performed by the Chemputer under full Chempiler control, and the program used, as well as a description of the process in prose, can be found in the SM. The automated synthetic procedure was started by automatically priming the tubes to the chemical reservoirs and then autocleaning the backbone with water, isopropanol, and dry diethyl ether. The system then continued autonomously through the whole synthesis of diphenhydramine hydrochloride without human intervention as follows. Diethyl ether and a small portion of bromobenzene were added to the magnesium, and the mixture was heated under reflux to initiate the Grignard reagent formation. After cooling, the remaining bromobenzene was added at a rate of 1 ml/min, and the mixture was again

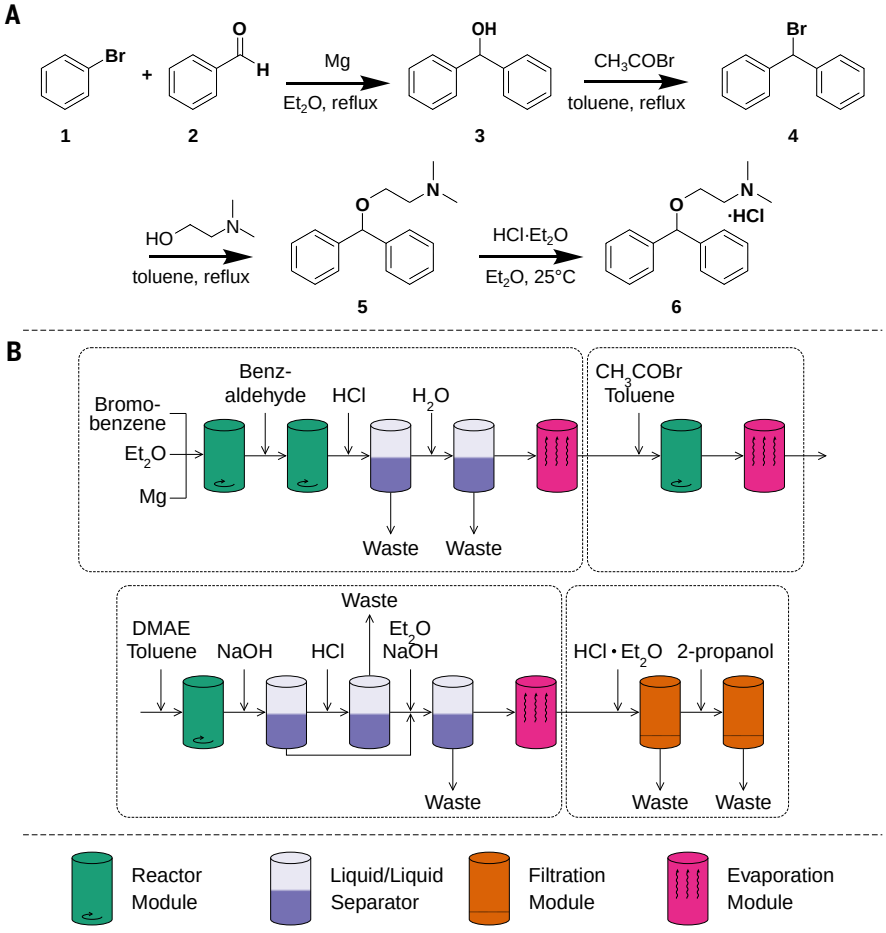


Fig. 4. Synthesis of diphenhydramine hydrochloride (6). (**A**) Modified synthetic route to diphenhydramine hydrochloride. Et, ethyl. (**B**) Sequence of unit operations required for the synthesis. The dotted boxes denote the four stages of the synthesis. DMAE, dimethylaminoethanol.

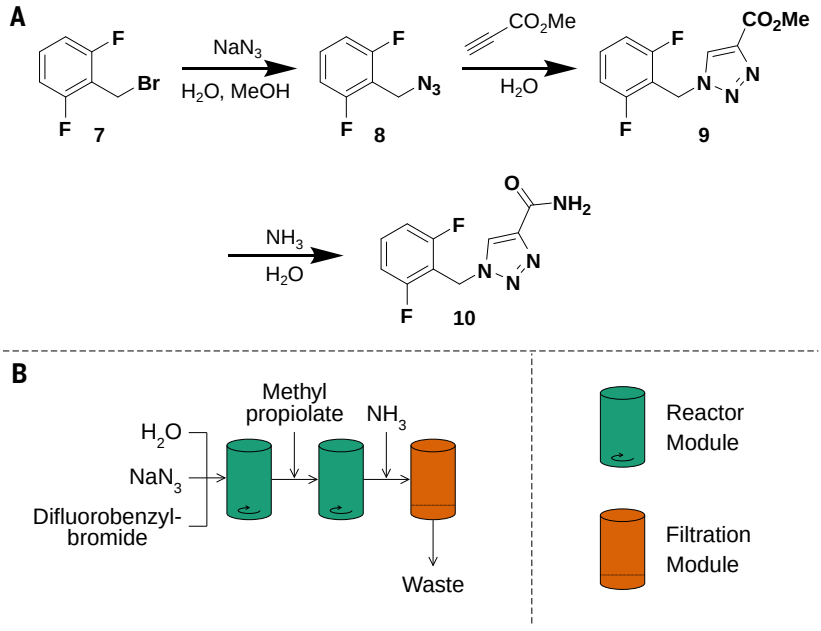


Fig. 5. Synthesis of rufinamide (10). (**A**) Synthetic route to rufinamide. Me, methyl. (**B**) Sequence of unit operations required for the synthesis.

heated to reflux. Using syringe pumps for moving material allowed us to precisely control addition rates and thus increase the reproducibility of synthetic protocols even further. Subsequently, a solution of benzaldehyde in diethyl ether was added at a rate of 1 ml/min, and the mixture was held at reflux for another 5 hours. As the platform presented herein is largely a proof of concept, no process analytical technology (PAT) has been implemented yet, so all reaction times were determined beforehand and hard-coded into the ChASM script. However, the modular architecture of the platform and control software should make adding PAT to future iterations of the platform straightforward. After quenching of the reaction with dilute HCl, the layers were separated, and the organic layer was washed with water and concentrated in vacuo as described earlier. The system then cleaned itself and directly proceeded with the bromination. To that end, the reactor was charged with acetyl bromide, the crude diphenylmethanol (**3**) was transferred from the rotary evaporator flask to the reactor with three portions of toluene, and the mixture was heated to reflux, all without human intervention. After a predetermined reaction time, the mixture was transferred to the rotary evaporator and evaporated to dryness. The subsequent Williamson ether synthesis was initiated in a similar fashion, and after a predetermined time, the reaction was quenched with aqueous sodium hydroxide. The system then automatically conducted an aqueous workup and concentrated the product in vacuo. Once again, the system cleaned itself, charged the jacketed filtration module with hydrochloric acid, and transferred the crude free base **5** to the filter with three portions of diethyl ether. To ensure smooth precipitation, the mixture was stirred vigorously and the free base solution was added very slowly. After the addition was completed, the off-white precipitate was collected by automatic filtration and recrystallized from isopropanol by using the heating and cooling capabilities of the jacketed filter. Drying at 60°C in a stream of argon for 1 hour yielded pure diphenhydramine hydrochloride, giving an isolated yield of 58% over four steps, or 87% per step on average. Although this is slightly less than the 68% overall achieved manually, the average yield per step (87% for automated versus 91% for manual) is comparable in our view. The platform performed the synthesis fully automatically in 77 hours (see movie S1), whereas the manual synthesis took 4 work days.

Next, we conducted an automated synthesis of the antiseizure drug rufinamide (**10**), a triazole derivative commonly prepared via a click reaction between the corresponding azide **8** and methyl propiolate (Fig. 5) (*21*). The synthesis began with an azide formation, for which the reactor was charged manually with 2,6-difluorobenzyl bromide (**7**); the remaining reagents were provided in bottles as described above. From this point on, unless explicitly stated otherwise, all described operations were performed by the Chemputer under full Chempiler control. An aqueous solution of sodium azide was added to the reactor to prepare the organic azide for the triazole click with methyl propiolate, which was performed inside the jacketed filtration module. Subsequent saponification with aqueous ammonia led to precipitation of the target compound. Filtration followed by three aqueous washes yielded pure rufinamide at 46% isolated yield, which was slightly better than the manual synthesis (38%). The automated synthesis took 38 hours. To demonstrate the power of the Chempiler software and the interoperability of the code, we then went on to run the same ChASM file on a "full-scale" platform equipped with slightly different hardware, which was connected in an entirely different way. The platform produced pure rufinamide in 44% yield without any problems or changes to the code (movie S2).

In the next synthesis, we prepared sildenafil, better known under the brand name VIAGRA (**17**), as shown in Fig. 6 (*13*). For this synthesis, we fitted the reactor with a heating block connected to the recirculation chiller, allowing us to cool as well as heat. From this point on, unless explicitly stated otherwise, all described operations were performed by the Chemputer under full Chempiler control. The reactor was cooled to 15°C and automatically charged with chlorosulfonic acid, thionyl chloride, and molten ethoxybenzoic acid. Chlorosulfonic acid is corrosive, so when writing the ChASM script we took great care to minimize contact times and enacted a strict cleaning regime. Chlorosulfonic acid and thionyl chloride also react violently with trace amounts of water, producing large volumes of gas. Therefore, the backbone was automatically flushed with dry diethyl ether and dried with a small amount of thionyl chloride before the reactor was charged. After a predetermined reaction time, the filtration module was charged with water and cooled to 0°C. The reaction mixture was then slowly dripped into the water, quenching the excess thionyl chloride and chlorosulfonic acid and precipitating the product (**12**), which was collected by automated filtration. The subsequent sulfonamide formation with *N*-methylpiperazine (**13**) in water was performed in the filtration module as well. Unfortunately, the sulfonamide (**14**) did not crystallize spontaneously, so a slurry of a small amount of product in water was added to seed the crystallization.

The industrial process for sildenafil uses *N*, *N*′-carbonyldiimidazole for the amide coupling in the next step. However, this reaction did not work in our hands, either manually or in automation, and thus we decided to use the acid chloride instead. The carboxylic acid (**14**) was thoroughly dried by flowing argon through the filter cake while at the same time heating the filtration module to 60°C, after which acid chloride formed with thionyl chloride in dichloromethane. The reactor module was charged with a solution of 4-amino-1-methyl-3-*n*-propyl-1*H*-pyrazole-5-carboxamide (**15**) in dichloromethane and triethylamine and cooled to 10°C. The crude acid chloride solution was then pumped from the filter module to the reactor, the reaction was quenched with water, the layers were separated, and the organic layer was dried over activated molecular sieves and evaporated to dryness, yielding amide **16** as an off-white solid. For the subsequent cyclization, the crude amide was transferred back to the reactor by dissolving it with a solution of potassium *tert*-butoxide in *tert*-butanol, and the mixture was heated at reflux for 8 hours. After the mixture was cooled to 10°C, the reaction was quenched with water, and the solution was transferred to the filter module. To induce precipitation of the sildenafil,

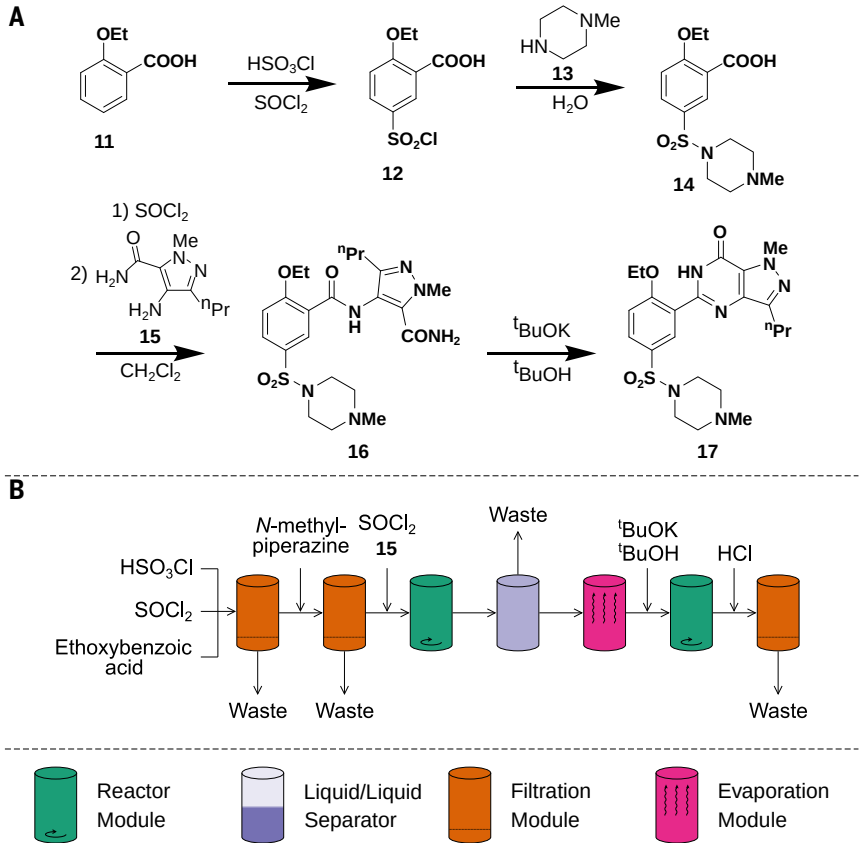


Fig. 6. Synthesis of sildenafil (17). (**A**) Synthetic route to sildenafil. ^{t}Bu, *tert*-butyl; ^{n}Pr, *normal*-propyl. (**B**) Sequence of unit operations required for the synthesis.

the mixture was neutralized with aqueous hydrochloric acid. After filtration, the solid was washed with water and dried under a stream of argon at 50°C, yielding sildenafil at 44% isolated yield over 102 hours (movie S3).

Outlook

The complete automation of all of synthesis is an ambitious objective, but this work makes a start toward that goal, as the Chemputer architecture presents a general abstraction of the process that works with traditional bench-scale techniques. The versatile programming language, the use of traditional and inexpensive labware (the total cost of the parts for the robotic modules, including nonstandard glassware, is less than $10,000 per system) (*22*), and the payoff in reproducibility after validation of the process mean that adoption could be straightforward. Initially, the synthesis of compounds will be validated reaction by reaction, but we imagine that eventually it will be possible to go straight from a reaction database to a chemical code that can run the platforms.

Materials and methods summary

The manual and automated syntheses of the three target molecules, as well as the platform and the control software, are described in detail in the SM, and this information has been deposited in Zenodo (*23*), along with the ChASM and GraphML code. Videos of the automated syntheses are available as movies S1 to S3. We will continue to update the ChASM code for the syntheses, and updates of this will be available to download (*24*). A brief summary of the syntheses is provided below.

Synthesis of diphenhydramine hydrochloride

The reactor module was charged manually with magnesium grit and dried by heating to 150°C under a stream of argon for 15 min. After cooling to room temperature (~25°C), diethyl ether and 2 ml of bromobenzene were added to the magnesium, and the mixture was heated to reflux for 20 min. After cooling below 25°C, 8.65 ml of bromobenzene was added at a rate of 1 ml/min, and the mixture was again heated to reflux for 20 min. Subsequently, a solution of benzaldehyde in diethyl ether was added at 1 ml/min, and the mixture was held at reflux for 5 hours. After quenching of the reaction with dilute HCl, the layers were separated, and the organic layer was washed with water and evaporated to dryness, yielding crude diphenylmethanol. After automatic cleaning of the system, the reactor was charged with acetyl bromide, and the crude diphenylmethanol was transferred from the rotary evaporator to the reactor with three portions of toluene. The mixture was heated to reflux for 4 hours and subsequently evaporated to dryness, yielding crude bromodiphenylmethane. The system was automatically cleaned once more, the reactor was charged with 2-(dimethylamino)ethanol and 10 ml of toluene, and the bromodiphenylmethane was transferred to the reactor with three portions of toluene. The mixture was heated to reflux for 20 hours. After cooling below 30°C, the reaction was quenched with aqueous sodium hydroxide. The layers were separated, and the organic layer was extracted with 2 M aqueous hydrochloric acid three times. Equimolar aqueous sodium hydroxide was added to the combined aqueous layers, and the mixture was extracted with diethyl ether three times. The combined etheric layers were evaporated to dryness, yielding crude diphenhydramine free base. The jacketed filter module was then charged with etheric hydrochloric acid, and the crude free base was slowly transferred to the filter with three portions of diethyl ether. The precipitate formed was collected by filtration, dried under a stream of argon, and recrystallized from isopropanol, yielding pure diphenhydramine hydrochloride as white crystalline powder.

Synthesis of rufinamide

The reactor was charged manually with 2,6-difluorobenzyl bromide. An aqueous solution of sodium azide was added to the reactor, and the mixture was heated to 75°C for 12 hours and subsequently transferred to the jacketed filter module. Neat methyl propiolate was added, and the mixture was heated to 65°C for 4 hours. An aqueous solution of ammonia was subsequently added, and the mixture was held at 75°C for an additional 12 hours, precipitating the target compound. Filtration followed by three aqueous washes yielded pure rufinamide as white crystalline powder.

Synthesis of sildenafil

The reactor was automatically charged with chlorosulfonic acid, thionyl chloride, and molten ethoxybenzoic acid. The mixture was stirred at 15°C for 30 min and then at room temperature for an additional 12 hours. Subsequently, the filtration module was charged with water and cooled to 0°C. The reaction mixture was slowly dripped into the water, quenching the excess thionyl chloride and chlorosulfonic acid and precipitating the product. The supernatant solution was removed, and the precipitate was washed with cold water, yielding 5-chlorosulfonyl-2-ethoxybenzoic acid as white powder. The wet solid remaining in the filter module was subsequently slurried in cold water, and *N*-methylpiperazine was added slowly. After 5 min, crystallization was initiated by adding a slurry of a small amount of product. The solid was collected by filtration, washed with cold water, and dried under a stream of argon at 50°C, yielding 2-ethoxy-5-(4-methyl-1-piperazinesulfonyl)-benzoic acid as a white powder. The dry carboxylic acid was slurried in dichloromethane and cooled to 5°C. Thionyl chloride and a catalytic amount of dimethylformamide were added, and the mixture was stirred for 5 hours at 25°C. Subsequently, the reactor module was charged with a solution of

4-amino-1-methyl-3-*n*-propyl-1*H*-pyrazole-5-carboxamide in dichloromethane and triethylamine and cooled to 10°C. The crude acid chloride solution was pumped from the filter module to the reactor, and the mixture was stirred for 16 hours at 25°C. After the reaction was quenched with water, the layers were separated, and the organic layer was dried over activated molecular sieves and evaporated to dryness, yielding 4-[2-ethoxy-5-(4-methyl-1-piperazinylsulfonyl)-benzamido]-1-methyl-3-propyl-1*H*-pyrazole-5-carboxamide as an off-white solid. For the subsequent cyclization, the crude amide was transferred back to the reactor by dissolving it with a solution of potassium *tert*-butoxide in *tert*-butanol, and the mixture was heated at reflux for 8 hours. After cooling to 10°C, the reaction was quenched with water, and the solution was transferred to the filter module. To induce precipitation of the sildenafil, the mixture was neutralized with aqueous hydrochloric acid. After filtration, the solid was washed with water and dried under a stream of argon at 50°C, yielding the sildenafil as white crystalline powder.

REFERENCES AND NOTES

1. J. Li *et al*., Synthesis of many different types of organic small molecules using one automated process. *Science* **347**, 1221–1226 (2015). doi: 10.1126/science.aaa5414; pmid: 25766227
2. P. J. Kitson *et al*., Digitization of multistep organic synthesis in reactionware for on-demand pharmaceuticals. *Science* **359**, 314–319 (2018). doi: 10.1126/science.aao3466; pmid: 29348235
3. C. Rougeot, H. Situ, B. H. Cao, V. Vlachos, J. E. Hein, Automated reaction progress monitoring of heterogeneous reactions: Crystallization-induced stereoselectivity in amine-catalyzed aldol reactions. *React. Chem. Eng.* **2**, 226–231 (2017). doi: 10.1039/C6RE00211K
4. F. Häse, L. M. Roch, C. Kreisbeck, A. Aspuru-Guzik, Phoenics: A Bayesian optimizer for chemistry. *ACS Cent. Sci.* **4**, 1134–1145 (2018). doi: 10.1021/acscentsci.8b00307; pmid: 30276246
5. A. Adamo *et al*., On-demand continuous-flow production of pharmaceuticals in a compact, reconfigurable system. *Science* **352**, 61–67 (2016). doi: 10.1126/science.aaf1337; pmid: 27034366
6. A. G. Godfrey, T. Masquelin, H. Hemmerle, A remote-controlled adaptive medchem lab: An innovative approach to enable drug discovery in the 21st century. *Drug Discov. Today* **18**, 795–802 (2013). doi: 10.1016/j.drudis.2013.03.001; pmid: 23523957
7. R. B. Merrifield, Automated synthesis of peptides. *Science* **150**, 178–185 (1965). doi: 10.1126/science.150.3693.178; pmid: 5319951
8. G. Alvarado-Urbina *et al*., Automated synthesis of gene fragments. *Science* **214**, 270–274 (1981). doi: 10.1126/science.6169150; pmid: 6169150
9. M. B. Plutschack, B. Pieber, K. Gilmore, P. H. Seeberger, The hitchhiker's guide to flow chemistry. *Chem. Rev.* **117**, 11796–11893 (2017). doi: 10.1021/acs.chemrev.7b00183; pmid: 28570059
10. O. J. Plante, E. R. Palmacci, P. H. Seeberger, Automated solid-phase synthesis of oligosaccharides. *Science* **291**, 1523–1527 (2001). doi: 10.1126/science.1057324; pmid: 11222853
11. P. D. Karp, Pathway databases: A case study in computational symbolic theories. *Science* **293**, 2040–2044 (2001). doi: 10.1126/science.1064621; pmid: 11557880
12. M. Baker, 1,500 scientists lift the lid on reproducibility. *Nature* **533**, 452–454 (2016). doi: 10.1038/533452a; pmid: 27225100
13. GraphML, http://graphml.graphdrawing.org/ [accessed 19 October 2018].
14. D. G. Brown, J. Boström, Analysis of past and present synthetic methodologies on medicinal chemistry: Where have all the new reactions gone? *J. Med. Chem.* **59**, 4443–4458 (2016). doi: 10.1021/acs.jmedchem.5b01409; pmid: 26571338
15. A. C. Bédard *et al*., Reconfigurable system for automated optimization of diverse chemical reactions. *Science* **361**, 1220–1225 (2018). doi: 10.1126/science.aat0650; pmid: 30237351
16. A. Adamo, P. L. Heider, N. Weeranoppanant, K. F. Jensen, Membrane-based, liquid-liquid separator with integrated pressure control. *Ind. Eng. Chem. Res.* **52**, 10802–10808 (2013). doi: 10.1021/ie401180t
17. M. O'Brien, P. Koos, D. L. Browne, S. V. Ley, A prototype continuous-flow liquid-liquid extraction system using open-source technology. *Org. Biomol. Chem.* **10**, 7031–7036 (2012). doi: 10.1039/c2ob25912e; pmid: 22858870
18. S. Eppel, T. Kachman, Computer vision–based recognition of liquid surfaces and phase boundaries in transparent vessels, with emphasis on chemistry applications. arXiv:1404.7174 [cs.CV] (6 November 2014).
19. D. J. Dale *et al*., The chemical development of the commercial route to sildenafil: A case history. *Org. Process Res. Dev.* **4**, 17–22 (2000). doi: 10.1021/op9900683
20. A. Ahmadi, M. Khalili, R. Hajikhani, N. Safari, B. Nahri-Niknafs, Anti-inflammatory effects of two new methyl and morpholine derivatives of diphenhydramine on rats. *Med. Chem. Res.* **21**, 3532–3540 (2011). doi: 10.1007/s00044-011-9891-y
21. R. D. Padmaja, K. Chanda, A short review on synthetic advances toward the synthesis of rufinamide, an antiepileptic drug. *Org. Process Res. Dev.* **22**, 457–466 (2018). doi: 10.1021/acs.oprd.7b00373
22. The total cost of the parts for robotic modules, including nonstandard glassware, is less than $10,000 per system, but this rises to $30,000 if the stirrers, evaporator, and chiller system are included, although these components might reasonably be expected to be found in most well-equipped synthesis laboratories.
23. G. Keenan, croningp/ChemputerSoftware: Chemputer first release, Zenodo (2018). doi: 10.5281/zenodo.1481731
24. Chemify; www.chemify.org.

ACKNOWLEDGMENTS

We thank M. Craven for help with the XDL development and H. Mehr, S. Zalesskiy, M. Symes, and R. Hartley for comments on the manuscript. **Funding:** We gratefully acknowledge financial support from the EPSRC (grants EP/H024107/1, EP/J015156/1, EP/K021966/1, EP/L015668/1, and EP/L023652/1) and the ERC (project 670467 SMART-POM). **Author contributions:** L.C. conceived the initial concept design approach and coordinated the team. P.J.K., G.A.-C., G.K., and L.C. developed the abstraction. L.C., T.H., and S.G. developed the fluidic backbone and initial versions of the pumps and valves, including the electronics and firmware. T.H. outlined the initial XDL with L.C. S.S. built the final versions of the rig, developed ChASM, and conducted the syntheses with help from J.W., A.A., and J.M.G. D.A. helped with the sildenafil automation. G.K. and G.A.-C. developed the software system with S.S. L.C. and S.S. wrote the manuscript. **Competing interests:** L.C. is the founder and director of DeepMatter Group PLC (deepmatter.io), which aims to digitize chemistry. **Data and materials availability:** All the software to run the Chemputer to synthesize the compounds described herein has been deposited in Zenodo (*23*).

SUPPLEMENTARY MATERIALS

www.sciencemag.org/content/363/6423/eaav2211/suppl/DC1
Materials and Methods
Supplementary Text
Figs. S1 to S58
Movies S1 to S3
Data S1 and S2

30 August 2018; accepted 12 November 2018
Published online 29 November 2018
10.1126/science.aav2211

Science

Organic synthesis in a modular robotic system driven by a chemical programming language

Sebastian Steiner, Jakob Wolf, Stefan Glatzel, Anna Andreou, Jaroslaw M. Granda, Graham Keenan, Trevor Hinkley, Gerardo Aragon-Camarasa, Philip J. Kitson, Davide Angelone and Leroy Cronin

Science **363** (6423), eaav2211.
DOI: 10.1126/science.aav2211originally published online November 29, 2018

Clear directions for a robotic platform

The chemistry literature contains more than a century's worth of instructions for making molecules, all written by and for humans. Steiner *et al.* developed an autonomous compiler and robotic laboratory platform to synthesize organic compounds on the basis of standardized methods descriptions (see the Perspective by Milo). The platform comprises conventional equipment such as round-bottom flasks, separatory funnels, and a rotary evaporator to maximize its compatibility with extant literature. The authors showcase the system with short syntheses of three common pharmaceuticals that proceeded comparably to manual synthesis.

Science, this issue p. eaav2211; see also p. 122