Cubesat Autonomicity

A Paradigm Shift in Cubesat Autonomicity

Clement Gama, Roy Sterritt, George Wilkie, Glenn Hawe
School of Computing
Ulster University
Jordanstown Campus, Newtownabbey, County Antrim BT37 0QB, United Kingdom
Gama-C1@ulster.ac.uk, R.Sterritt@ulster.ac.uk, Fg.Wilkie@ulster.ac.uk, Gi.Hawe@ulster.ac.uk

Abstract — This paper presents an autonomic (self-managing) model, which can be used in cubesat (nano satellite) development to assist in designing cubesats that are S**elf-C**onfiguring, **Self-H**ealing, **Self-O**ptimizing and **Self-P**rotecting – (Self-CHOP). In this paper we explore – in-depth the four autonomic characteristics Self-CHOP from which cubesat space missions will benefit in terms of mission cost reduction, data integrity, spacecraft health status, and above all, the reduction of space debris (junk). Self-Configuring (the dynamic auto-system configuration) defines self-setup based on current internal and external environmental factors; it applies predefined rules of self-preservation to alter the spacecraft configuration parameters depending on the Cubesat Autonomic Capability Model level the spacecraft implements. Implementing self-optimising is the performance assessment of the cubesat with regards to its mission goals; it recalibrates its instruments based on current orbital factors and properties for optimal function. The increase in space debris has increased the probability of collisions between natural space debris and man-made satellites or satellite-to-satellite crashes. Self-protection is the proposed solution to circumvent such collisions which would otherwise cause more space debris (Kessler Syndrome). Some faults experienced by on-board instruments (sensors) can be fixed by the autonomic manager responsible for the spacecraft welfare. Self-healing is implemented to solve software faults and other non-mechanical faults. It requires a spacecraft that is self-aware and has normal operation parameters pre-defined which are used to detect sensor operations outside the normal range. In this paper we show that self-managing spacecraft means less man-mission ratios therefore missions are cheaper to run, and Self-CHOP increases the number of successful missions.

Keywords - autonomic computing; autonomicity; apoptosis; cubesat; capability model.

I. INTRODUCTION

Cubesats, also known as nano satellites weighing between 1.33kg for a "1U 10cm cube" and 10 kilograms for a "6U 10 x 20 x 30cm" and above, were originally designed to provide experimental space platforms (with limited functionality) [1] [2] for organisations such as universities, space agencies, high schools and private companies [3].

Since the inception of the CubeSat standard at California Polytechnic State University in collaboration with Stanford University [3], it has evolved over the years to become centre-stage in the space industry [4]. Typical applications for cubesats include earth observation science, in-orbit communications and services, and remote sensor observations [5]. The National Aeronautics and Space Administration (NASA) has been deploying cubesats into deep space [6], a purpose that was not anticipated by Professor Puig-Suari and Professor Twiggs [3] at the beginning of the cubesat revolution in 1999 [6]. NASA has had their cubesat Launch Initiative which has enabled small payloads built by schools, universities and private companies to be deployed in space and conduct scientific experiments [7].

The original intent for cubesats was to deploy them in Low Earth Orbit (LEO) which meant their lifespan in space was to be very short – some cubesats could be in orbit for about 6 weeks to over

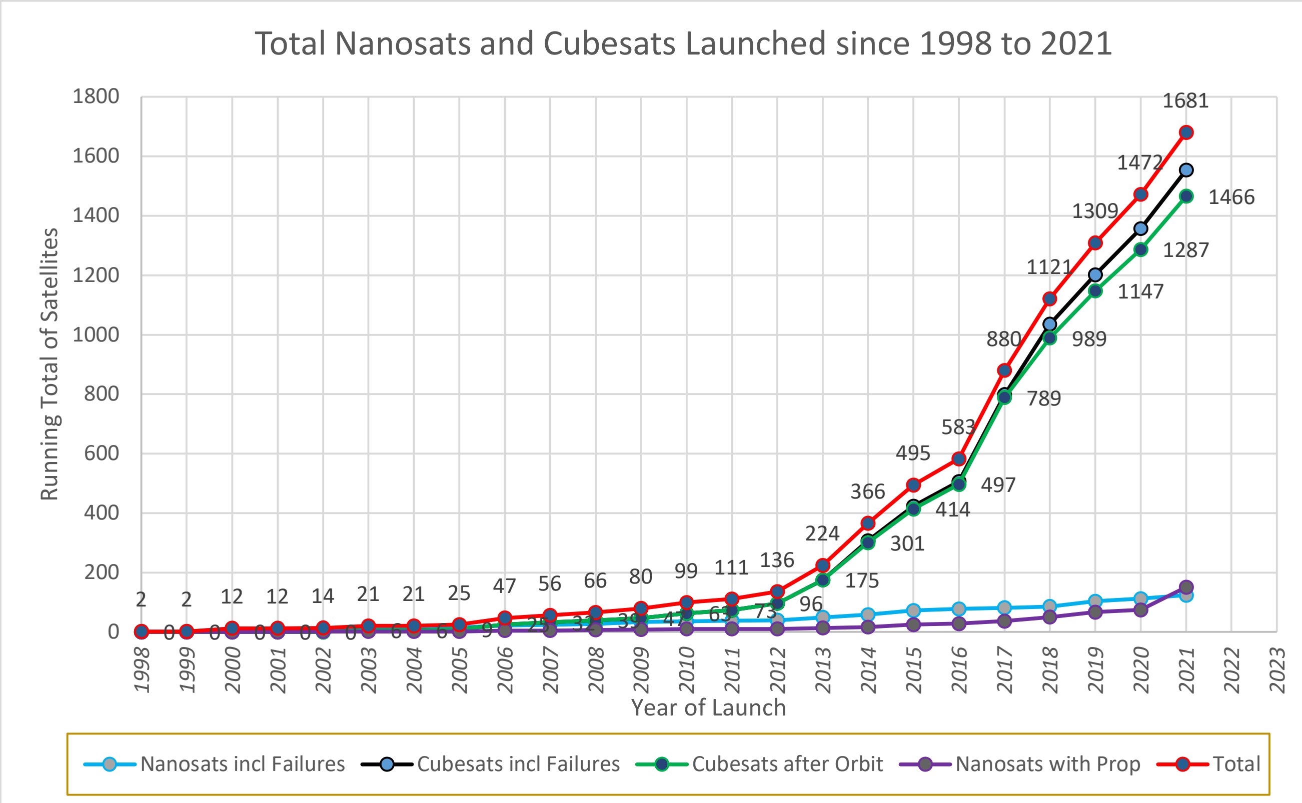

Figure 1shows the number of Cubesats, Pocketcubes, Nanosats, Picosats, Femtosats, Thinsats deployed since 1998. Data extracted from "Erik Kulu, Nanosats Database, https://www.nanosats.eu"

1 year. Cubesats are mostly built from low cost Commercial Off-The-Shelf (COTS) components which may be hardened for space use [8]. COTS components may be cheaper compared to large monolithic satellite components, but they are still expensive for most high school and some university projects. In large spacecraft the payload consists of one or more instruments or sensors that perform scientific experiments and collect data [4]. The structure of the generic spacecraft comprises Propulsion Systems, Guidance and Navigational Control (GNC) – being a combination of Orbit Determination and Control Systems (ODCS) and Attitude Determination and Control Systems (ADCS) [4] - Electric Power Supply (EPS) subsystems [9], Thermal Control [10], Radio Communication Systems with high gain antennas and Command and Data Handling systems [4].

However, cubesats, can have equivalent or similar systems but at miniaturized sizes e.g., they cannot have high gain antennas due to the physical size restriction, mass restriction and the lack of enough electrical power to drive the antenna systems on board [11][12]. During the past 2 decades, cubesats have revolutionised the space industry by providing an entry point into space exploration [13][14] by allowing would-be space companies to experiment and demonstrate their ideas without the huge investment required when designing and implementing space missions [15].

The revolution of cubesats has over time made it necessary to make them intelligent and make them last longer in space. The initial strategy for using cubesats was to make them disposable, and therefore there was less need to keep them in orbit longer than necessary, and besides there was no feasible propulsion system that could be fitted within the physical size constraints of 10cm cubes etc.

This paper seeks to demonstrate that cubesats can be designed to be smart at varying levels. Hence the introduction of the Cubesat Autonomic Capability Model (CACM) which is presented as the solution to the on-going problem of space debris due to defunct satellites and debris that has come from collisions of spacecraft with natural debris or with other satellites.

II. THE PROBLEM

An increase in space commercialisation has been the result of small satellite business explosion due to the revolutionary low-cost and quick roll-out time schedules that small satellites bring to the space industry [16]. The rate at which space objects are increasing is exponential given that constellations require and operate multiple – from a few to thousands - spacecraft as part of their network [16], which also means more debris will also increase exponentially. For example SpaceX's Starlink constellation - (initially intended to have 12,000 satellites within this decade may be increased to 42,000) - which provides wireless internet even in remote areas that traditional communications are scarce if not completely unavailable [17]. Starlink has the potential to create more space debris than any other constellation deployed before. Other constellations will easily double or triple that number, creating a space environment that might not be sustainable in future [16]. Whilst new satellites of all sizes including cubesats are being deployed so increasingly and thus bringing huge benefits to humanity, they are also the greatest source of disruption to the space industry if space governance is not taken seriously and therefore managed properly [18]. Over a hundred launches every year bring over 200 spacecraft of various sizes into space, thereby cluttering space [19]. Space is becoming overcrowded especially at the LEO range since about 70% of all cubesats are orbiting at this altitude.

Figure 1 shows a graphical representation of number of cubesats and other small satellites e.g. picosats, femtosats, nanosasts etc. deployed in space since 1998 up until April 2021 [20]. It is almost 3500 cubesats flying in LEO and collisions become more and more likely as this number is ever increasing. According to the nanosats database [20] there are only 74 (as of this writing) out of the 3500 cubesats that have a propulsion system, those that can be manoeuvred and possibly avoid collisions. The rest are flying in space waiting for their orbit to decay naturally and cannot be moved out of harm's way, therefore are the greatest threat to active satellites. Collisions are unavoidable as long as these satellites no matter how small or big cannot be manoeuvred out of harm's way. That is just referring to other satellites and trackable space debris, not mentioning objects too small to track.

Space collisions are of three categories, namely:-

- Satellite-to-satellite
- Satellite-to-existing debris
- Debris-to-debris

Satellite-to-satellite

Collisions between two or more artificial satellites is the main source of space debris in LEO. If monolithic satellites and cubesats had the intelligence to avoid collisions, this is where debris additions could be reduced. Most of the time cubesats are the greater threat to larger satellites because, as shown in Figure 1, most of them lack manoeuvre features. Cubesats can also cause collisions with other cubesats due to either atmospheric drag changing their altitude faster than expected or due to perturbations causing cubesats to change from their initial orbit. In the space industry, it is now routine to dedicate resources to monitor and protect spacecraft from collisions. Such resources work on manually manoeuvring their spacecraft to avoid devastating collisions with space debris or other satellites [18].

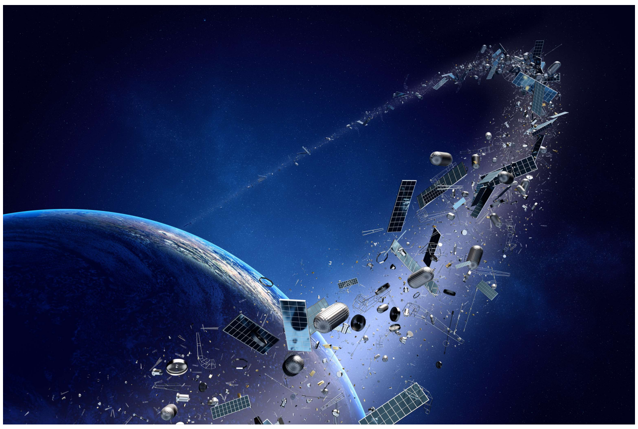

Figure 2 Space debris orbiting around earth - picture supplied by istockphoto.com

Satellite-to-Debris

There are thousands of debris already flying at all altitude levels that are being monitored by NASA's Orbital Debris Program Office (ODPO). ODPO estimates to be about half a million marblesize space debris flying at high velocity (8km/s [21]), and about 100 million 1 millimetre-size or smaller objects travelling at dangerously high velocity [22]. It is almost impossible to predict where every piece of space "junk" / debris will be at any given time. So therefore, space missions are always at risk of collisions with these pieces flying at high velocity in LEO, Medium Earth Orbit (MEO) and Geostationary Orbit (GEO) [22]. The debris pieces, though some are very small, can inflict serious damage to any spacecraft due to their high velocity [23]. According to the European Space Agency (ESA) there are about 129 million space debris objects flying in space with sizes ranging from 1 millimetre and above [23]. As of January 2019, on average ESA performed one manoeuvre per year on their satellites to avoid existing space debris [18].

Debris-to-debris

Debris accumulation has reached critical levels, and as predicted by Kessler [24] such that more debris is being created / added to the debris cloud due to collision within the cloud itself. This type of collision simply adds to the existing debris by merely existing. This is the most difficult type to avoid because it would require manual removal of the debris to stop the Kessler syndrome. Space agencies, however, have started putting mitigation plans in place, e.g. burning up all rocket fuel in each rocket module to prevent fuel explosions later on [21]. Other space agencies like SpaceX have designed and deployed re-usable rockets e.g. the Falcon 9 – their rocket stages are not left flying in space, they are brought back to earth for future deployment [26].

The other problem with space missions in general is the cost associated with every mission from design to deployment, to maintenance. In a typical space mission involving monolith spacecraft, satellites send their data (instrument data & craft navigation and health status data) back to ground station for analysis. The data analysts and engineers send back commands to the craft for the next tasks to be performed by the spacecraft. This exercise has a high cost because the number of missions has increased over the years, and since each mission requires a certain number of dedicated staff, the number of mission engineers has had to

Year	Mission	No. of spacecraft	No. of people	Current People / spacecraft	Mission Objectives
2001	Wilkinson Microwave Anisotropy Probe (WMAP)	1	4	4:1	To provide a more detailed look at temperature differences in the cosmic microwave background [25]
2000	Iridium Constellation	66	200	3:1	Launched 66 satellites into polar orbit at altitudes of about 400 miles. Each of six orbital planes, separated by 30 degrees around the equator, will contains eleven satellites [26].
2000	GlobalStar	48	100	2:1	Low Earth Orbit (LEO) satellite constellation for satellite phone and low-speed data communications extension of terrestrial cellular systems [26].
2007	New Millennium Program (NMP) Space Technology 5 (ST5)	3	12	-	Advanced the technology of miniaturizing smart and powerful electronic gadgets by building and testing three small satellites, also known as micro-sats.
2012	Magnetotail Constellation (MC)	30 - 40	120 - 160	-	Magnetotail Constellation is designed to explore plasma transport and energy conversion processes over spatial sizes ranging from the distance to the Sun to the size of low energy particle gyro-orbit [27]

Table 1 shows ratio of human staff to space missions increase. Engineers have to monitor the spacecraft health status, its navigation, and its intended mission goal. In many cases, missions require 24 hours 7 days a week management which further increases the cost of the mission in its lifetime [25].

Another pressing issue is space communication delays experienced when sending commands from ground stations to satellites in deep space. This increases the risk of missions failing because it takes a very long time for the ground operators to receive and process the data from deep space before they know what is happening out there, and by that time a lot could have gone wrong [27].

These problems - lack of propulsion in most LEO satellites, collisions causing debris, costs for running high personnel staffed missions and delayed communications between ground stations and the spacecraft - with space missions require a solution to address them in a quick and cost-effective manner.

I. AUTONOMICITY

Autonomicity defines how a system self-manages - using Self-CHOP - its own internal subsystems (self-awareness) and also is aware of its own environment [28]. An autonomic system follows its mission with minimal or no interference from external sources in terms of configuration and or control [29]. It can also be seen as a reflexive reactive system and quite spontaneous [30].

The original vision of autonomicity was inspired by biological systems in that they should always self-destruct (apoptosis) unless they receive a "stay alive signal" [28]. In an autonomic system, sensors and actuators are core, they are the equivalent of a biological nervous system. Through sensor data acquisition, a system's behaviour can be monitored and controlled [30].

International Business Machines (IBM) created a four-part model for an autonomic system consisting of an autonomic manager which itself

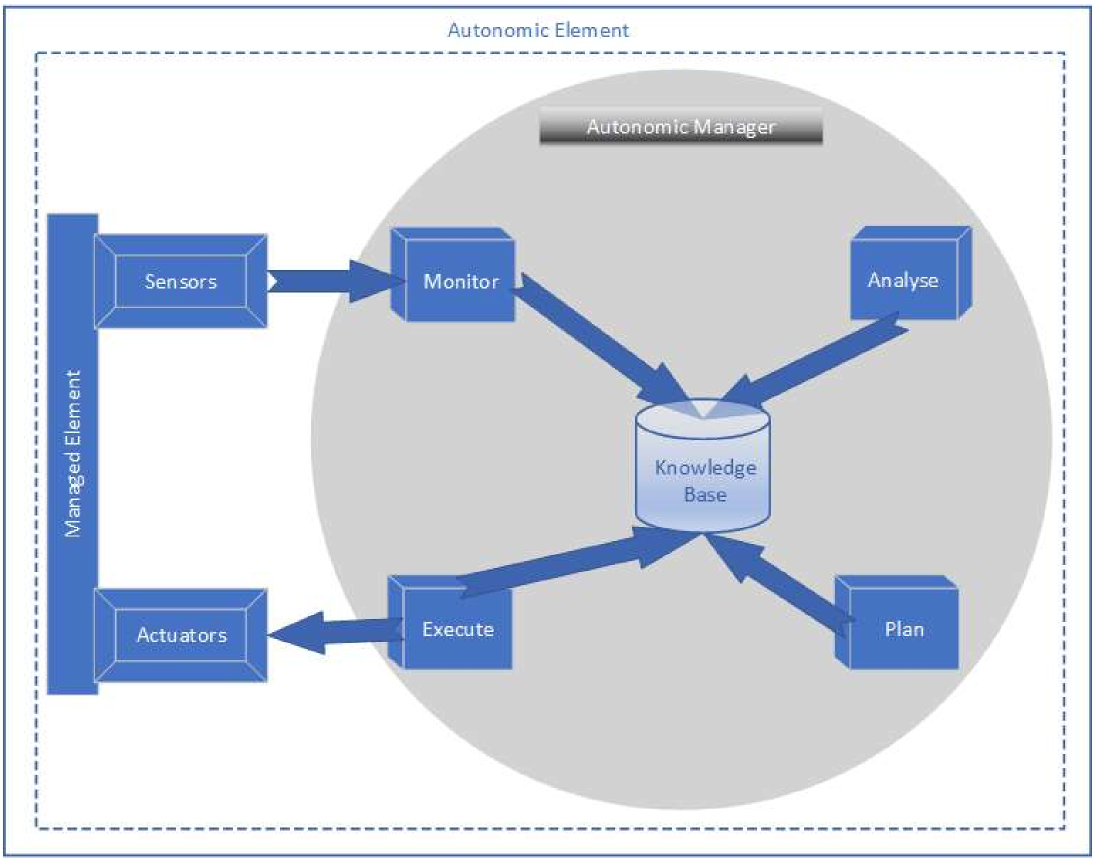

Figure 3 IBM Autonomic Manager showing interface with managed element through sensors and actuators.

comprises monitoring, analysing, planning, execution and a knowledgebase (MAPE-K) phases [31]. Outside of the MAPE-K loop, there is a managed element which consists of sensors and actuators [32]. The monitoring subsystem interfaces with the sensors to monitor the element, and the execution subsystem interfaces with the actuators (effectors) to either change configuration parameters or perform certain system manoeuvres [32]. The system uses the knowledgebase during analysis to plan the next course of action based on historical events.

Given the young and evolving nature of the cubesat industry and hence the lack of autonomic control in cubesats, the purpose of this paper is to present an autonomic model that – if implemented – will see cubesats change their longevity, usefulness and cost effectiveness.

II. Proposed Debris Clean-up Solutions

Active research shows there are a variety of solutions proposed to address the debris problem caused by either satellites that have come to the end of life or defunct spacecraft that is floating in orbit with no control mechanism to de-orbit them [16]; usually these satellites would have run out of fuel or their instruments would be defunct. One of the proposed solutions is to deorbit satellites near their end-o-life using built-in propulsion, and the others are classified into two types: On-Orbit Satellite Servicing (OOSS) and Active Debris Removal (ADR). The disadvantage of OOSS is the high cost associated with it, since it works by refuelling old satellites, repowering them and tweaking the configuration to enable them to continue working after they have been considered dead [33]. The OOSS high cost due to the high technology required to revive these defunct satellites is still cheaper than building a new satellite and launching it [33].

Active debris removal involves changing the debris attitude and trajectory which eventually deorbits the object to burn up in the atmosphere [33].

Other solutions include an Inflatable De-Orbit Device (iDod) which greatly shortens the de-orbit time of cubesats by creating drag for the cubesat which slows it down [34] enough to stay in orbit for less than the 25 years required by the Inter-Agency Space Debris Coordination Committee (IADC) [35]. The advantage of this solution is that it is cost effective because of its small volume and light weight sail [34]. Other variants of inflatable de-orbit devices include pillow and or balloon [36].

Some devices clean-up space using high-power lasers to completely annihilate debris that is a few centimetres in size. They work by changing an object's orbit direction so that it deorbits and burns up in the atmosphere [37].

Zero-Debris Space Construction is a group of debris clean-up satellites which can change their orbits or have tethers to use when they rendezvous with space debris to slow it down or change the debris trajectory. These clean-up satellites have to be kept at an idle orbit waiting to move when a debris object is detected to possibly collide with another or with a spacecraft [38].

III. Cubesat Autonomic Capability Model

Here we present one possible solution to the overcrowding problem of space especially Low Earth Orbit (LEO). As already discussed in previous paragraphs, cubesats are the greatest threat now to other satellites and they are the most vulnerable to other space debris due to the lack of propulsion on the smaller sized Thinsats, Picosats, etc.

The Cubesat Autonomic Capability Model (CACM) was inspired by Autonomic Computing (AC) as defined by the IBM 2001 Autonomic Maturity Model [39] and by Capability Maturity Model Integration (CMMI). The IBM maturity model implements 5 levels of autonomic capabilities.

Level 1 is basic and there is no autonomic features, all MAPE-K plans, execution instructions and monitoring are manual [40]. Level 2 is the managed level – management software is in place to provide consolidation and automation of IT tasks. In level 3 individual components monitor, correlate, and analyse their environment and suggest possible actions [40]. Level 4 of the model implements all features in level 3, but in addition the components take action with minimal human intervention in their processes [40]. The highest level – level 5 – has components fully integrated, dynamically and collectively manged by business rules [40]. There is no human intervention in the workflows, but it does not preclude intervention by engineers if they deem it necessary e.g. to make business rule changes.

The other source of inspiration for the CACM is CMMI which is a collection of best practices meant to help businesses continually improve their business processes [41]. CMMI's framework structure is used to create models, training materials and appraisals. Within the framework constellation models exist and are created using goals and practices [41]. CMMI aims to reduce complexity, redundancy resulting from using disparate capability models and make processes more cost effective [42].

Integrated into the CACM design also is the Autonomy Levels Framework [43] which is suitable for moving vehicles as opposed to the IBM framework which is perfectly suitable for systems with a lot of electric power, a lot of computing power and are stationary [44]. This framework is a customised version of autonomic computing catering for low powered environments like in spacecraft where resources are quite limited [45].

The CACM is formulated to be product specific unlike CMMI which is organisation specific [41]. CACM is designed to address the lack of autonomicity in cubesats given their size, mass, cost of development and launch. If implemented, this model can aid developers in their systemic design and development of cubesats which can help incorporate autonomic computing from design to completion. CACM can be used to validate cubesat features and autonomic capabilities as a standard.

Implementing CACM can result in safer, intelligent, low cost, self-destructing cubesats and increase in the probability of mission successes. Cubesats designed with and incorporating CACM will be safer for the space environment because they will implement apoptosis which ensures a system will autonomically self-destruct unless it receives a "stay-alive signal". Self-destruction will mean the cubesat de-orbits itself and burns up in the upper atmosphere upon entry, thereby cleaning up what would otherwise be another space debris.

Also, if cubesats have propulsion, CACM compliant cubesats will Self-Protect (part of Self-*) by avoiding potential collisions with other spacecraft or space debris. Such cubesats can also protect themselves against space radiation e.g., by shutting themselves down during solar storms. Implementing Self-Protect is crucial to the success of cubesat missions because it extends the lifespan of the cubesat, thereby allowing it to perform its scientific tasks for much longer than would otherwise be, thus reducing mission costs.

1. *CACM Core Features & Functions*

Cubesats implementing CACM will contain most of the following features in order to be viable for space operations [4]; the complexity of the core features is always custom designed according to the nature of the space mission. CACM follows the MAPE-K autonomic manager structure, and therefore will monitor every subsystem, analyse its status, plan course of action and execute the instructions required to achieve the intended outcome.

1) *Hardware*

Hardware starts with the chassis which houses all components and also is used as a radiation shield and heat radiator. Most cubesat frames are custom made since the payload is always custom, therefore there is less use for preformed chasses [4].

2) *Electrical Power Supply (EPS)*

Electrical power is supplied by space hardened batteries and solar panels for charging the batteries. EPS controls power distribution, storage, regulation and other power control units [4]. The system will prioritize power distribution according to power constraints. In case of power shortage, only critical instruments will receive power, the rest will be shutdown.

3) *Attitude Determination & Control System (ADCS)*

The cubesat will monitor and determine its own attitude, and in a constellation configuration will collaborate with other constellation members. Attitude will be changed to achieve intended mission goals.

4) *Orbit Determination and Control (ODC)*
It determines and controls orbit parameters, semi-major axis, semi-minor axis, eccentricity, inclination, right ascension of the ascending node, argument of periapsis and true anomaly.

5) *Communications*
This is the interface between the cubesat and ground station. The cubesat downlinks its scientific data and health status data to ground stations. It also sends ground station commands to the cubesat. This is a crucial element in the life of the cubesat and the mission as a whole. Without data communication the mission is dead, so data transmission has to be robust and resilient.

6) *Ground Station Systems (GSS)*
Although the ground station is not part of the cubesat, but it is the main control centre for the cubesat mission, and as such the systems running in the ground stations will be design specific for each mission. Ground stations plan new missions, change current mission parameters and upload new missions. They also receive and analyse spacecraft data.

7) *Autonomic Manager (AM)*
The autonomic manager will coordinate all subsystems within the cubesat and will serve as the main point of contact between the cubesat and ground station. All the autonomic subsystem modules will be controlled by the manager thus ensuring the latest system health status is coordinated throughout the spacecraft. The manager will fully implement the MAPE-K loop. This is where the level of autonomicity the cubesat implements is setup, monitored and managed.

8) *Health Monitoring*
Scientific instruments have to have a health reporting feature in order to be properly managed. Instruments' health status is important in determining if the cubesat is still viable or not after faults have occurred in sensors and payload instruments. If all scientific instruments are defunct, and any attempts to revive them fail then control is handed over to De-Orbit Control to end the mission. Internal sensors will be used to determine the health status of the cubesat, and that data will be relayed down to ground station, but also used to determine the course of action if the cubesat is fully autonomic.

9) *De-Orbit Control*
This will enable a "death by default" (apoptosis) feature which will ensure the cubesat de-orbits whenever it loses communication with the ground stations for an extended period of time to be determined at mission design. The de-orbit function will also be used at the end of the cubesat mission to change its trajectory so that it re-enters the atmosphere and burns up.

10) *Constellation Capability*
Autonomic cubesats will participate in constellations and behave in the same manner as when they are on single craft missions in terms of their autonomicity. All constellation members must be fully autonomic, which means they must implement the highest level of autonomicity as single cubesats.

CUBESAT AUTONOMIC CAPABILITY MODEL (CACM)	
CACM Level	Autonomic Cubesat Level Description
Level 1 **Cubesat Managed from Ground Station**	• Fully managed from ground station – ground station calculates visibility time slots and plans communication sessions within visibility times. • Mission type is fixed – the cubesat mission parameters are hard coded. • Limited on-board capability – No propulsion • Always transmitting telemetry data • Constellation: cannot participate in a constellation.
Level 2 **Ground Station (GS) & Cubesat Shared Control**	• Basic autonomicity – cubesat reports its health status to ground station • Default functions: Apoptotic feature, minimal propulsion • Mission is pre-scheduled, mission operations on-board. • Transmits data to ground station on a schedule. • Constellation: cannot participate in a constellation.
Level 3 **Single Cubesat Full Autonomic Control**	• Single Cubesat Full Autonomicity – implements MAPE-K and Self-CHOP - ground station can intervene if deemed necessary • Mission is pre-scheduled, mission operations on-board. • Transmits data to ground station during visibility timeslots • Kill switch autonomously (apoptotic) executed and or by ground station – goes through a check and verify process before executing the kill switch • Mission goals can be adapted mid-mission
Level 4 **Basic Constellation Management**	• ONLY applies to Constellations • Constellation cubesat missions implement Self-CHOP • Execution of goal-oriented mission operations on-board. • Individual members have to be at CACM Level 3 - autonomic internal systems operations. • Send health status to ground station and constellation. • Allows ground station to veto kill-switch execution.
Level 5 **Full Autonomic Constellation Management**	• ONLY applies to Constellations • Goal-oriented mission operations on-board. • Can self-re-initialize OS and internal systems – no human intervention • Sends health status to ground stations. • Only receives new mission from ground station. • Kill switch notification with error details • Ground Station can always intervene as and when necessary

Table 2 shows the summary structure of the Cubesat Autonomic Capability Model

IV. CACM LEVELS

In level 1, the cubesat monitoring subsystem polls sensors for data and heartbeats at fixed intervals and whatever data is returned from the sensors is sent to ground station. The data could be erroneous or invalid, but since there is no data validation onboard the cubesat, the system does not check the data for errors.

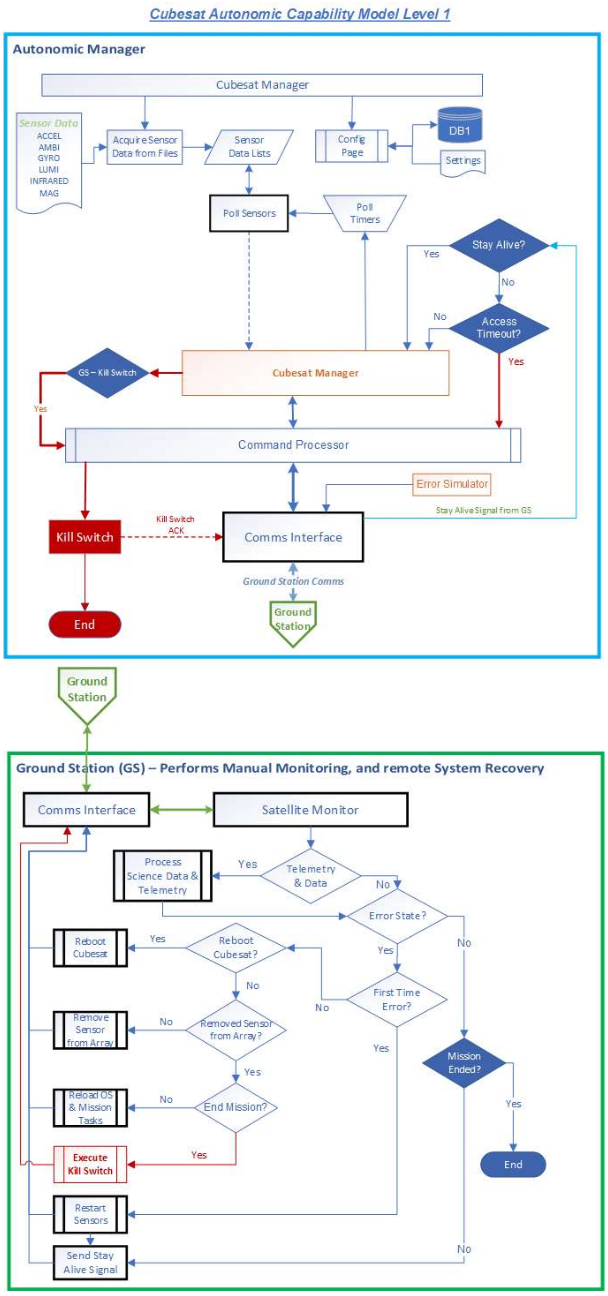

Figure 4 shows the block diagram of a cubesat at CACM level 1 with its interaction with ground station.

The cubesat is continuously sending data down to ground station – at least trying to – even if the ground station is out of sight.

Data sent includes telemetry data (heartbeats) and science instrument data. Ground station is fully responsible for running the mission, and therefore all data validation is performed on the ground station. Ground station – based on the data received from the cubesat – creates execution plans and sends commands to the cubesat to perform error recovery. If after several attempts to recover from an error condition the faulty sensors cannot be fixed, then ground station will decide whether to deorbit the cubesat or not depending on its viability. Viability mainly depends on how many sensors are still fully functional at any point in time.

A cubesat on level 2 sends health status data at times when the ground station is in the line of sight. Science data and health data validation is performed at high level (very basic) – it only checks if data being received is within expected ranges only. When sensors become faulty or die, the system only restarts the sensors, and if error recovery fails, the system shuts down those sensors and the mission continues. If all sensors die, the cubesat will re-initialize itself in an attempt to perform error recovery. If all attempts fail, then the kill switch will be executed by the cubesat or ground station might intervene and perform other diagnostics. Ground station has full control of the cubesat even though it has limited autonomicity onboard.

In level 3 a cubesat implements all the features of autonomicity that a single spacecraft can. CACM levels are cumulative, i.e. each new level adds on the previous level's features. A level 3 cubesat will refresh its mission goals from onboard storage, recalibrate its sensors and perform full error recovery attempts. It will reset faulty sensors until the maximum number to resets is reached and then the whole cubesat will be reset if the errors are still not cleared. If the maximum cubesat reboots have been reached, the kill switch will be executed unless ground station intervenes. If, and when a cubesat in this level detects possible or imminent collisions with other spacecraft or debris, it will navigate around the potential object using its propulsion system.

In level 4 – which only applies to constellations – a cubesat must be at level 3 and constellation aware. It should be able to announce that it is joining the constellation and what features and functions it possesses and can announce its exit from the constellation. It basically communicates with the constellation manager(s). The individual cubesat health data and science data is sent to the manager for downlink to ground station. In case the sensors on an individual constellation member become faulty, the cubesat will remove itself from the constellation to perform error recovery similar to level 3. If it recovers from the error condition, it re-joins the constellation and performs whatever tasks are assigned by the constellation coordinator.

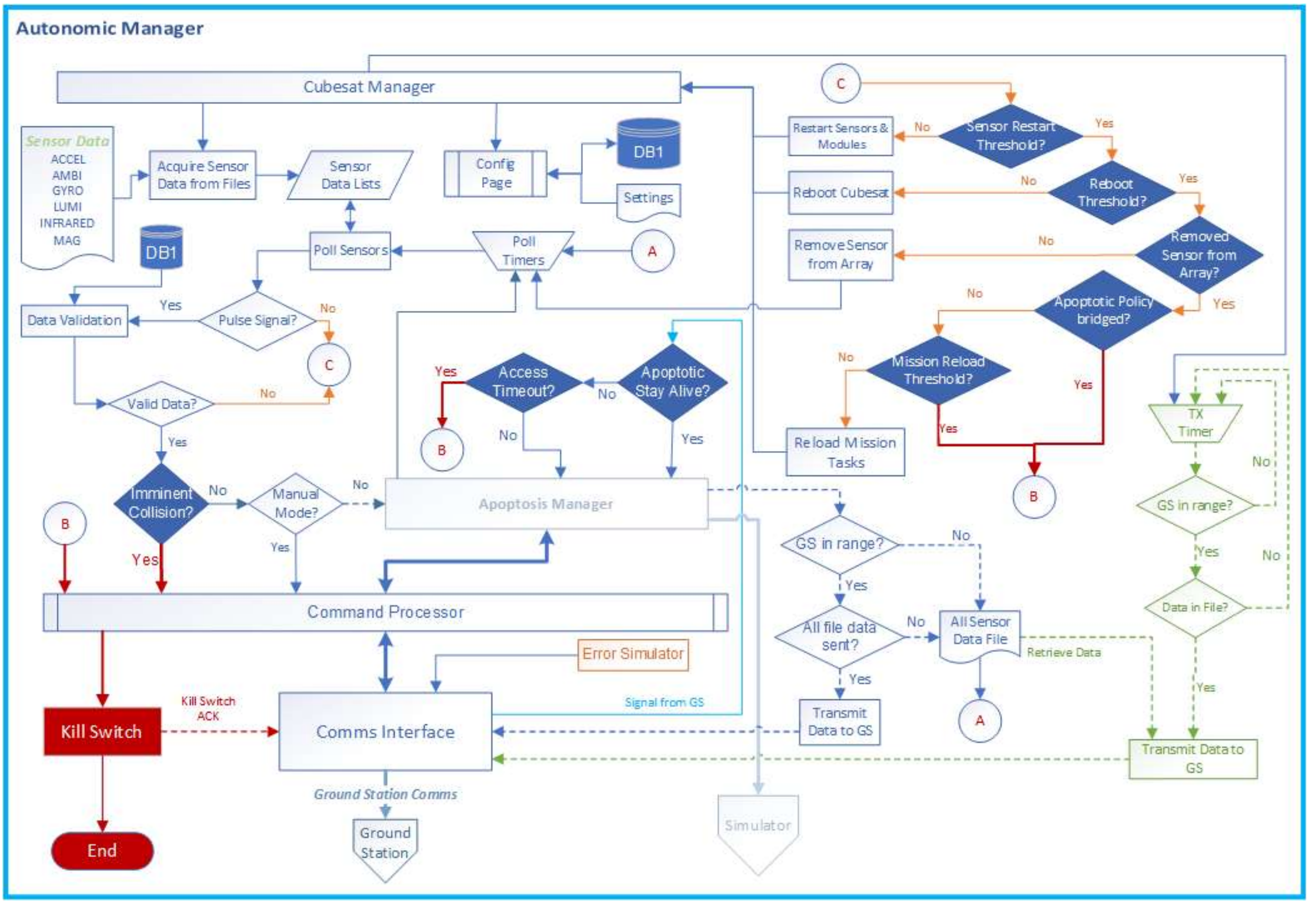

Figure 5 shows a block diagram of cubesat at CACM level 3

In addition to the features and capabilities of level 4, in level 5 the cubesats are grouped by similar or related payload instrumentation. This creates redundancy groups which become useful when one cubesat goes faulty since others in the group can easily take over the roles of the faulty cubesat which might have to exit the group temporarily to perform error recovery. The constellation manager is responsible for the temporary or permanent assignment of the tasks to other members of the same redundancy group. These groups need to move in formation within their environment whilst keeping safe distances amongst themselves. The constellation manager is tasked with protecting the constellation and as such monitors the health status and payload data from the other members with the intention to remove any faulty cubesat from the constellation if it does not self-remove. If the constellation manager becomes faulty, a vote is held to decide which of the healthy members can take over the role of management. If member instruments and sensors become faulty, the same error recovery process performed at level 3 is followed, and if error recovery fails, the kill switch is executed to save and protect the rest of the constellation.

V. Conclusions

We have presented autonomic computing, its history and origins as defined by IBM in 2001. A review of the IBM maturity model has been discussed and it partly formed the basis for the CACM that this presentation has focused on. We also revisited CMMI as another maturity and capability model used as inspiration in creating and designing the CACM.

CACM is targeted at cubesat developers who intend to make their cubesats intelligent, safer, cheaper and longer lasting in missions. CACM comprises 5 levels similar to the CMMI and IBM maturity models. Each level adds a layer of sophistication on top of the previous level's features and functions of the core cubesat features and functions. In summary, CACM level 1 does not implement any autonomicity, level 2 is partial autonomic features, and level 3 implements full autonomic functions for a single cubesat. Levels 4 and 5 only apply to constellations, but all members of any constellation must already be in CACM level 3 to ensure the constellation is efficient and better protected.

If CACM is implemented, it can save companies a lot of costs when it comes to mission running and management post launch. It could, however, increase design and cubesat building costs due to the complexity and the number of sensors and monitoring required to enable full autonomic features.

References

[1] N. Cheeks, "From the Chief," *NASA Goddard Tech Transf. News*, vol. 11, no. 2, pp. 1–24, 2013.

[2] CalPoly San Luis Obispo, C. L. Initiative, NASA Advanced Exploration Systems Division, S. C. and N. Division, N. J. P. Lab, and NOAA, "CubeSat 101: Basic Concepts and Processes for First-Time CubeSat Developers."

[3] CalPoly, "Cubesat design specification," *CubeSat Program, Calif. Polytech. State …*, vol. 8651, no. June 2004, p. 22, 2009, doi: http://dx.doi.org/10.1155/2015/681901.

[4] A. Poghosyan and A. Golkar, "CubeSat evolution: Analyzing CubeSat capabilities for conducting science missions," *Prog. Aerosp. Sci.*, vol. 88, no. November 2016, pp. 59–83, 2016, doi: 10.1016/j.paerosci.2016.11.002.

[5] H. G. Lewis, B. S. Schwarz, S. G. George, and H. Stokes, "AN ASSESSMENT OF CUBESAT COLLISION RISK."

[6] JPL NASA, "Mars Cube One (MarCO)." http://www.jpl.nasa.gov/cubesat/missions/marco.php.

[7] A. Heiney, "CubeSat Launch Initiative - Introduction," 2020. http://www.nasa.gov/content/cubesat-launch-initiative-introduction (accessed Apr. 01, 2021).

[8] A. R. Aslan *et al.*, "Development of a LEO communication CubeSat," in *RAST 2013 - Proceedings of 6th International Conference on Recent Advances in Space Technologies*, 2013, pp. 637–641, doi: 10.1109/RAST.2013.6581288.

[9] A. Ali, M. R. Mughal, H. Ali, and L. Reyneri, "Innovative power management, attitude determination and control tile for CubeSat standard NanoSatellites," *Acta Astronaut.*, vol. 96, no. 1, 2014, doi: 10.1016/j.actaastro.2013.11.013.

[10] G. S. Aglietti, C. W. Schwingshackl, and S. C. Roberts, "Multifunctional structure technologies for satellite applications,"

Shock Vib. Dig., vol. 39, no. 5, pp. 381–391, Sep. 2007, doi: 10.1177/0583102407077397.

[11] P. Pergola, "Small satellite survey mission to the second Earth moon," *Adv. Sp. Res.*, vol. 52, no. 9, 2013, doi: 10.1016/j.asr.2013.07.043.

[12] A. G. C. Guerra, F. Francisco, J. Villate, F. Aguado Agelet, O. Bertolami, and K. Rajan, "On small satellites for oceanography: A survey," *Acta Astronautica*, vol. 127. 2016, doi: 10.1016/j.actaastro.2016.06.007.

[13] CalPoly, "Cubesat design specification rev. 13," *CubeSat Program, Calif. Polytech. State …*, vol. 8651, p. 22, 2009, [Online]. Available: http://wdww.cubesat.org/images/developers/cds_rev13_final2.pdf.

[14] X. Wu, F. Wu, and J. Zhao, "Motion deblurring for space-based imaging on sandroid cubesats using improved genetic algorithm," *Procedia Comput. Sci.*, vol. 80, pp. 2327–2331, 2016, doi: 10.1016/j.procs.2016.05.431.

[15] C. Underwood, S. Pellegrino, V. J. Lappas, C. P. Bridges, and J. Baker, "Using CubeSat/micro-satellite technology to demonstrate the Autonomous Assembly of a Reconfigurable Space Telescope (AAReST)," *Acta Astronaut.*, vol. 114, pp. 112–122, 2015, doi: 10.1016/j.actaastro.2015.04.008.

[16] H. Lewis *et al.*, "SENSITIVITY OF THE SPACE DEBRIS ENVIRONMENT TO LARGE CONSTELLATIONS AND SMALL SATELLITES."

[17] H. Pettit, "What will the Starlink internet speed be and how much will it cost?," 2021.

[18] ESA, "ESA - Automating collision avoidance," *Safety & Security*, 2019. https://www.esa.int/Safety_Security/Space_Debris/Automating_collision_avoidance (accessed May 13, 2021).

[19] C. Bonnal, J. M. Ruault, and M. C. Desjean, "Active debris removal: Recent progress and current trends," *Acta Astronautica*, vol. 85. pp. 51–60, 2013, doi: 10.1016/j.actaastro.2012.11.009.

[20] E. Kulu, "Nanosats Database | Constellations, companies, technologies and more," Jan-. https://www.nanosats.eu/ (accessed Apr. 01, 2021).

[21] E. Gregersen, "space debris | Facts, Removal, & Research | Britannica," *Encyclopaedia Britannica*. https://www.britannica.com/technology/space-debris (accessed May 13, 2021).

[22] NASA ODPO, "ARES | Orbital Debris Program Office." https://www.orbitaldebris.jsc.nasa.gov/ (accessed May 13, 2021).

[23] R. Pool, "Space debris: the scrapheap in the sky | E&T Magazine," *Engineering & Technology Magazine*, 17-Mar-2021. https://eandt.theiet.org/content/articles/2021/03/space-debris-the-scrapheap-in-the-sky/ (accessed May 13, 2021).

[24] E. Levin, J. Pearson, and J. Carroll, "Wholesale debris removal from LEO," *Acta Astronaut.*, vol. 73, pp. 100–108, 2012, doi: 10.1016/j.actaastro.2011.11.014.

[25] W. Truszkowski, L. Hallock, J. Karlin, J. Rash, and G. Michael, "Autonomous and Autonomic Systems," p. 289, 2010, doi: 10.1007/b105417.

[26] SpaceX, "SpaceX - Falcon 9." https://www.spacex.com/vehicles/falcon-9/ (accessed May 13, 2021).

[27] H. Spence, T. Moore, and A. Klimas, "The Magnetotail Constellation Global Dynamics of the Structured Magnetotail," no. October, 2004.

[28] R. Sterritt and M. G. Hinchey, "From Here to Autonomicity : Self-Managing Agents and the Biological Metaphors that Inspire Them," *Proc. Integr. Des. Process Technol. Symp. (IDPT 2005)*, pp. 143–150, 2005.

[29] T. O. Eze, R. J. Anthony, A. Soper, C. Walshaw, A. Computing, and U. Kingdom, "A Technique for Measuring the Level of Autonomicity (LoA) of Self-managing Systems," no. c, pp. 8–13, 2012.

[30] R. Sterritt and M. Hinchey, "Adaptive

reflex autonomicity for real-time systems," *Innov. Syst. Softw. Eng.*, vol. 5, no. 2, pp. 107–115, 2009, doi: 10.1007/s11334-009-0090-7.
[31] E. Dancy and U. Markova, "Autonomic computing in Canadian academia , Part 1 Autonomic computing projects in the Toronto Centre for Advanced Studies Project : Software Tuning Panels for Autonomic Control Research problem," pp. 1–10, 2008, doi: 10.1002/spe.
[32] M. C. Huebscher and J. a McCann, "A Survey of Autonomic Computing—Degrees, Models, and Applications," *ACM Comput. Surv.*, vol. 40, no. 3, pp. 1–28, 2008, doi: 10.1145/1380584.1380585.
[33] Staff Writers, "Active Debris Removal: A Necessary Evil," *Space Safety Magazine*, 2013. http://www.spacesafetymagazine.com/space-debris/debris-removal/active-debris-removal-evil/ (accessed May 11, 2017).
[34] D. Maessen, E. van Breukelen, B. T. C. Zandbergen, and O. K. Bergsma, "Development of a Generic Inflatable De-Orbit Device for CubeSats," *Proc. 58th Int. Astronnautical Congr.*, pp. 1–11, 2007, [Online]. Available: http://lr.tudelft.nl/fileadmin/Faculteit/LR/Organisatie/Afdelingen_en_Leerstoelen/Afdeling_SpE/Space_Systems_Eng./Expertise_areas/Space_propulsion/Research/Thermal_thrusters/doc/Presentation.pdf.
[35] H. Hakima and M. R. Emami, "Assessment of active methods for removal of LEO debris," *Acta Astronaut.*, vol. 144, pp. 225–243, Mar. 2018, doi: 10.1016/j.actaastro.2017.12.036.
[36] R. Ash and J. Laub, "CubeSat Deorbit Devices Midterm Report Abstract :," pp. 1–13, 2011.
[37] S. Shen, X. Jin, and C. Hao, "Cleaning space debris with a space-based laser system," *Chinese J. Aeronaut.*, vol. 27, no. 4, pp. 805–811, Aug. 2014, doi: 10.1016/j.cja.2014.05.002.
[38] H. Sahara, "Evaluation of a satellite constellation for active debris removal," *Acta Astronaut.*, vol. 105, no. 1, pp. 136–144, 2014, doi: 10.1016/j.actaastro.2014.08.026.
[39] IBM, "Autonomic Computing White Paper: An Architectural Blueprint for Autonomic Computing," *IBM White Pap.*, no. June, p. 34, 2005, doi: 10.1021/am900608j.
[40] IBM, "Autonomic Computing Toolkit," *User's Guide*, 2004. https://www.ibm.com/developerworks/autonomic/books/fpu0mst.htm#ToC_16 (accessed Jan. 15, 2018).
[41] CMMI Product Team, "CMMI for Development, Version 1.3 (CMU/SEI-2010-TR-033). Software Engineering Institute.," 2010. [Online]. Available: http://resources.sei.cmu.edu/asset_files/TechnicalReport/2010_005_001_15287.pdf.
[42] L. Greiner, "Capability Maturity Model Integration (CMMI) Definition and Solutions | CIO," *Tutorial Capability Maturity Model Integration (CMMI) Definition and Solutions*, 2007. http://www.cio.com/article/2437864/process-improvement/capability-maturity-model-integration--cmmi--definition-and-solutions.html (accessed Jan. 15, 2018).
[43] H.-M. Huang, K. Pavek, J. Albus, and E. Messina, "Autonomy Levels for Unmanned Systems (ALFUS) Framework: Safety and Application Issues," *Proc. SPIE*, vol. 5804, pp. 439–448, 2005, doi: 10.1117/12.603725.
[44] G. T. McWilliams *et al.*, "Evaluation of autonomy in recent ground vehicles using the autonomy levels for unmanned systems (ALFUS) framework," *Proc. 2007 Work. Perform. Metrics Intell. Syst. - Permis '07*, pp. 54–61, 2007, doi: 10.1145/1660877.1660884.
[45] H.-M. Huang, E. Messina, and J. Albus, "Autonomy Levels For Unmanned Systems (ALFUS) Volume II : Framework Models," *Framework*, vol. II, no. December, 2007.